*open*FT V10.0 for UNIX Systems

Installation and Administration

Edition December 2006

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion of this manual. Your feedback helps us optimize our documentation to suit your individual needs.

Fax forms for sending us your comments are included in the back of the manual.

There you will also find the addresses of the relevant User Documentation Department.

Certified documentation according to DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

This manual is printed on paper treated with chlorine-free bleach.

Contents

1 Preface

The *open*FT product range transfers and manages files
  – automatically,
– securely, and
– cost-effectively.

The reliable and user-friendly transfer of files is an important function in a high-performance computer network. Most corporate topologies today consist of networked PC workstations, which are additionally linked to a mainframe or Unix server. This allows much of the processing power to be provided directly at the workstation, while file transfer moves the data to the mainframe for further processing there as required. In such landscapes, the locations of the individual systems may be quite far apart. Fujitsu Siemens Computers offers an extensive range of file transfer products - the *open*FT product range - for the following system platforms:

- BS2000/OSD®
- Solaris™(SPARC®/Intel™), LINUX®, AIX®, HP-UX®, OSF1(Tru64)
- Microsoft® Windows XP™, Windows Server 2003™
- OS/390 resp. z/OS (IBM®)

1.1 Brief description of the product

*open*FT for UNIX systems is the file transfer product for systems with the UNIX operating system.

All *open*FT products from Fujitsu Siemens Computers intercommunicate via *open*FT protocols (originally: FTNEA protocols), which were standardized by Siemens. Since a number of FT products from other software vendors also support these protocols, many interconnection options are available.

When used in combination with *open*FT-FTAM, *open*FT also supports the FTAM file transfer protocol (File Transfer Access and Management) standardized by ISO (International Organization for Standardization). This makes it possible to interconnect with even more systems from other vendors whose file transfer products support the same standard.

When used in combination with *open*FT-FTP, *open*FT also supports the FTP protocol. This makes it possible to interconnect with other ftp servers.

With the integrated FTAC function, *open*FT offers extended admission and access protection (FTAC stands for **F**ile **T**ransfer **A**ccess **C**ontrol).

1.2 Target group and objectives of this manual

This manual contains the information which is needed by *open*FT and FTAC administrators of UNIX systems for their work and which is not included in the User Guide.

For general information on file transfer and file management, you will also need the User Guide. Further literature is listed in the references.

To understand this manual, it is useful to have a knowledge of the UNIX-based operating systems.

The manual covers Sun Solaris systems as well as portings to other UNIX platforms. The operating system-dependent differences are described in detail in the Release Notices supplied on the respective product CD.

1.3 Concept of *open*FT for UNIX systems manuals

The complete description of *open*FT and its optional components comprises four manuals. The description is divided among the manuals as follows:

- *open*FT for UNIX systems - Installation and Administration

  The system administrator manual is intended for FT and FTAC administrators. It describes:
  - the installation of *open*FT and its optional components,
  - the operation, control and monitoring of the FT system and the FTAC environment,
  - the administration commands for FT and FTAC administrators and
  - important CMX commands.

- *open*FT for UNIX systems - Enterprise File Transfer in the Open World

  The user manual is intended for the *open*FT user and describes:
  - the basic functions of the *open*FT product family,
  - the conventions for file transfers to computers running different operating systems,
  - details on implementing FTAM,
  - the *open*FT user commands,
  - the *open*FT-Script commands,
  - the BSFT interface,
  - the messages of the different components.

- *open*FT for UNIX systems and Windows systems - Program Interface

  This manual is intended for C programmers and describes the C program interface on UNIX systems and Windows systems.

- *open*FT for UNIX systems and Windows systems - *open*FT-Script Interface

  This manual is intended for XML programmers and describes:
  - the *open*FT-Script commands
  - the XML statements for the *open*FT-Script interface

Many of the functions described in the manuals are also available in the *open*FT graphical interface. A detailed online help system that describes the operation of all the dialogs in the graphical interface is supplied together with the graphical interface. The online help system also contains a complete description of the *open*FT commands.

1.4 Changes since the last version of the manual

This section describes the changes in *open*FT V10.0 for UNIX systems compared to *open*FT V8.1 for UNIX systems.

Partner addressing

Partners can be managed via a partner list. This is done using the new commands *ftaddptn*, *ftremptn*, *ftshwptn* and *ftmodptn*.

Extended support for character sets and character codes (CCS)

*open*FT supports Unicode, i.e. it is also possible to read in Unicode files using the *open*FT Explorer and exchange such files with partners as of *open*FT V10.

Support for IP V6

Partners as of *open*FT V10 can be addressed via IPv6 addresses.

Support for the ftp protocol

*open*FT supports file transfer and file management with ftp servers. If *open*FT is used as the remote ftp server then it is also possible to use FTAC functions as well as preprocessing and postprocessing.

Extended directory management

Using the new commands *ftcredir*, *ftmoddir* and *ftdeldir*, it is possible to create, rename and delete remote directories.

New script interface: *open*FT-Script

*open*FT-Script is an XML-based interface which offers flow control and context management as well as the FT functions. The new commands *ftscript*, *ftcans*, *ftdels*, *ftshws* and *ftshwact* are used to administer *open*FT-Script.

Other changes

- New commands *ftcanr* and *ftshwr* for the management of requests. These commands replace the *ftc* and *fti -q* commands which are to be discontinued.
- New commands *ftmodo* and *ftshwo* for setting and displaying the operating parameters. These commands replace the *fta* and *fti -p* commands which are to be discontinued.
- New commands *ftcrek* and *ftdelk* for creating and deleting key pair sets. These commands replace the functionality of the *fta* command which is to be discontinued.
- New commands *ftstart* and *ftstop* for starting and stopping the asynchronous *open*FT server. These commands replace the functionality of the *fta* command which is to be discontinued.
- New command *ftsetjava* to set the link to the Java executable and output information about Java executables.
- All traces are now prepared using the *fttrace* command. The command has been extended to permit this.
- Messaging has been modified (new message numbers/return codes, new texts).
- Remote administration is also possible via the graphical user interface.

1.5 Notational conventions

The following notational conventions are used throughout this manual:

`typewriter font`
: typewriter font is used to identify entries and examples.

italics
: In running text, names, variables and values are indicated by italic letters, e.g. file names, instance names, menus, commands and command options.


: indicates notes

Additional conventions are used for the command descriptions, see page 69.

1.6 README files

Information on any functional changes and additions to the current product version can be found in product-specific README files.

You will find the README files on the product CD under with name *readme.txt*. resp. *liesmich.txt.* They are no onger installed automatically.
You can view these files using an editor or print them out on a standard printer.

1.7 Current information on the Internet

Current information on the *open*FT family of products can be found on the World Wide Web under *http://www.fujitsu-siemens.com/openft.*

2 Tasks of the administrator

This chapter describes the most important administration tasks to be performed when running *open*FT. You can administer *open*FT both via the graphical interface (i.e. the desktop) and by using commands. The following options are available:

– Functions and commands that only the administrator may use (e.g. start *open*FT or delete log records),

– Functions and commands that are accessible to both the user and the administrator, but where the administrator is allowed to do more than the user (e.g. modify admission sets).

The tasks of the administrator include:

– Setting operating parameters[1) 2)]
– Starting and stopping *open*FT[1) 2)]
– Administering the request queue[1)]
– Viewing and deleting log records[1)]
– Administering admission sets and FT profiles[1)]
– Diagnostic options, e.g. switching the trace for error diagnostics on and off[1) 2)]
– Creating and administering instances in order to use *open*FT in the cluster
– Creating key pair sets [1)] and making a current public key available to the partner systems. This enables the local system to be authenticated by the partner.
– Obtaining the public keys of partner systems and suitably storing them in the local system so that the partner systems can be authenticated by the local system.

The administration functions marked with [1)] can also be executed via the ´graphical interface (i.e., the desktop), provided an X terminal or corresponding emulation is available. More information on the graphical interface can be found in the manual on "*open*FT V8.1 for UNIX systems" and in the online help installed with the graphical interface.

The administration functions marked with [2)] can also be performed via an SNMP management station.

Who is the *open*FT administrator?

*open*FT can only be administered under the login name *root*.

Who is the FTAC administrator?

The FTAC administrator manages admission sets and admission profiles. Both the *open*FT administrator and the FTAC administrator can manage logging.

Following a new installation, the *open*FT and FTAC administrators are identical (*root*). The FTAC administrator is identified by the fact that only his or her admission set is privileged. You can transfer this property to another login name by using the *ftmoda* command. This is useful, for example, if someone other than the system administrator is responsible for data security.

Depending on how (i.e., under which login name) the FTAC administrator has been set up, he or she will have different privileges and options, as explained below:

- Retention of *root* as the FTAC administrator or transfer of these privileges to another login name with *root* authorization:
  Every other login name (or user ID) with *root* authorization (i.e. UID=0) is also an FTAC administrator. Furthermore, the FTAC administrator has extended privileges (see the sections “Assigning privileges to FT profiles” on page 37 and “Saving the FTAC environment” on page 37).
- Transfer to a login name (or user ID) without *root* authorization (UID not equal to 0):
  The *open*FT administrator may no longer manage any admission sets and admission profiles. The FTAC administrator does not have extended privileges.
- Both the *open*FT administrator and the FTAC administrator can manage logging.

2.1 Setting the operating parameters

The following parameters are available for controlling the operation of *open*FT. You can specify these parameters by means of the *ftmodo* command:

- The instance identification of the local *open*FT instance.
- The maximum number of asynchronous requests that *open*FT should process simultaneously (CONN-LIM).
- The upper limit for the length of blocks to be transferred.

  Following the installation of *open*FT/*open*FT-FTAM, the maximum block length is set to 65535 characters.
- The scope for protocols during *open*FT operation.
- The length of the RSA key to be used for encryption purposes.
- The code table that should be used by default for local text files.

You can view the current values of the parameters for *open*FT with the *ftshwo* command.

You can also view and change the current operating parameters via the graphical interface. To do this, open the *Operating Parameters* window by selecting the appropriate menu item in the *Administration* menu. You will find a detailed description of each function in the online help for the graphical interface.

Tips for performance control

When specifying the value for CONN-LIM, you must consider the following points:

- A low value means that fewer FT requests can be run concurrently, but also implies that the performance of other applications will not be noticeably degraded on your processor.
- A high value means that a high volume of FT requests can be processed within a short period of time, but that the performance of other applications will be degraded on your processor.

2.2 Administering code tables

A code table defines a character set (Coded Character Set, CCS for short) and the coding of these characters in the file. A CCS is assigned a name of up to 8 characters in length via which the CCS can be addressed.

As FT administrator, you can use the *ftmodo -ccs* command to set a standard CCS for *open*FT. In addition, you are still able to set your own 8-bit CCS.

The standard CCS is used for all FT requests. However, users can set a different CCS in the *ft-/ncopy* request and in the *open*FT Editor.

The following CCSs are supplied with *open*FT as standard:

Name of the CCS	Meaning
ISO88591 to ISO8859A and ISO8859D to ISO8859F	for the ASCII tables ISO8859-1 to ISO8859-A and ISO8859-D to ISO8859-F
ISO646	for the international 7-bit ASCII table
ISO646DR	for the German 7-bit ASCII reference version
EDF041 to EDF04A and EDF04D to EDF04F	for the EBCDIC tables DF04-1 to DF04-10 and DF04-13 and DF04-15
EDF03IRV	for the international 7-bit EBCDIC table defined by FSC
EDF03DRV	for the German 7-bit EBCDIC table defined by FSC
UTF16	for Unicode with UTF-16 coding (platform-specific endian)
UTF8	for Unicode with UTF-8 coding
UTFE	for Unicode with the UTF-E coding defined by FSC
UTF16LE	for Unicode with UTF-16 coding (little-endian)
UTF16BE	for Unicode with UTF-16 coding (big-endian)
UTFEIBM	for Unicode with the UTF-EBCDIC coding defined by IBM
CP1252	for the ANSI character set with the Euro symbol defined by Microsoft (see above)
IBM1047	for the OpenExtensions EBCDIC character set defined by IBM

Name of the CCS	Meaning
CP850	for the OEM character set defined by Microsoft
IBM037	for the US/Canada EBCDIC character set defined by IBM
IBM500	for the International EBCDIC character set defined by IBM

Creating a user-defined CCS

If you are an openFT administrator, you can create your own CCS (Coded Character Set). To do this, you must create a text file which is stored in the *sysccs* subfolder of the *open*FT instance. The CCS name corresponds to the name of this file.

The text file must have the following structure:

– The first line starts with a '#'.

  The second character defines the code class (I for ISO8859, E for DF04).

  As of the third character, a space-terminated decimal number follows describing the code variant. The remainder of the line contains a comment which characterizes the code contained.

– The second line contains an alphabetic character which can at present only have the value 'S'. 'S' stands for single-byte code, i.e. a character is always 1 byte in length.

– The third line contains three numbers.

  The first number is a 4-digit hexadecimal number. This defines the substitution character to be used if a Unicode character cannot be mapped to the code.

  The second number is currently always '0'.

  The third number is a decimal number which defines the number of code pages that follow. It currently always has the value '1'.

– The following lines define the code pages and have the following structure:

  – The first of these lines contains the number of the code page in the form of a two-digit hexadecimal number.

– All the subsequent lines contain the mapping of the characters for the codes to be defined to UTF-16 in the form of a 4-digit hexadecimal number. The values are arranged in 16 lines, each of which contains 16 4-digit hexadecimal numbers with no spaces.

Example for EDF041

#E1 Encoding file: df04-1, single-byte															
S															
003F	0 1														
00															
0000	0001	0002	0003	0080	0009	0081	007F	0082	0083	0084	000B	000C	000D	000E	000F
0010	0011	0012	0013	0085	000A	0008	0087	0018	0019	0088	0089	001C	001D	001E	001F
008A	008B	008C	008D	008E	0086	0017	001B	008F	0090	0091	0092	0093	0005	0006	0007
0094	0095	0016	0096	0097	0098	0099	0004	009A	009B	009C	009D	0014	0015	009E	001A
0020	00A0	00E2	00E4	00E0	00E1	00E3	00E5	00E7	00F1	0060	002E	003C	0028	002B	007C
0026	00E9	00EA	00EB	00E8	00ED	00EE	00EF	00EC	00DF	0021	0024	002A	0029	003B	009F
002D	002F	00C2	00C4	00C0	00C1	00C3	00C5	00C7	00D1	005E	002C	0025	005F	003E	003F
00F8	00C9	00CA	00CB	00C8	00CD	00CE	00CF	00CC	00A8	003A	0023	0040	0027	003D	0022
00D8	0061	0062	0063	0064	0065	0066	0067	0068	0069	00AB	00BB	00F0	00FD	00DE	00B1
00B0	006A	006B	006C	006D	006E	006F	0070	0071	0072	00AA	00BA	00E6	00B8	00C6	00A4
00B5	00AF	0073	0074	0075	0076	0077	0078	0079	007A	00A1	00BF	00D0	00DD	00FE	00AE
00A2	00A3	00A5	00B7	00A9	00A7	00B6	00BC	00BD	00BE	00AC	005B	005C	005D	00B4	00D7
00F9	0041	0042	0043	0044	0045	0046	0047	0048	0049	00AD	00F4	00F6	00F2	00F3	00F5
00A6	004A	004B	004C	004D	004E	004F	0050	0051	0052	00B9	00FB	00FC	00DB	00FA	00FF
00D9	00F7	0053	0054	0055	0056	0057	0058	0059	005A	00B2	00D4	00D6	00D2	00D3	00D5
0030	0031	0032	0033	0034	0035	0036	0037	0038	0039	00B3	007B	00DC	007D	00DA	007E

2.3 Starting and stopping *open*FT

By default, *open*FT (i.e. the asynchronous *open*FT server) is started automatically at system startup. It is possible to deactivate automatic start-up via the graphical user interface (*Administration/Operational Parameters*).

If *open*FT is not started, only synchronous requests are executed. Asynchronous requests are stored in the request queue. Furthermore, no further requests are accepted from partner systems.

After being started, *open*FT executes both asynchronously issued requests as well as file transfer requests issued on the remote system.

You can start and stop *open*FT via the graphical interface (i.e. the desktop) with the *Administration/Start asynchronous server* or *Administration/End asynchronous server* functions or via the *ftstart and ftstop* commands.

2.4 Setting the protection bit for newly created files

You can set the protection bit value for new files created on reception to a value that restricts the file access rights for the owner, the group members and for other users.

You may modify the standard protection bit setting with the *umask* command. To ensure that the protection bit value is properly set when *open*FT is started, you should activate the command line *umask 027* in the startup file for the standard instance *std*. This startup file is located under */var/openFT/std/etcinit/openFTinst*.

2.5 Switching the language interface

The language is not queried during installation. Instead, the *LANG* environment variable of the administrator installing *open*FT is evaluated and set as the default language. This value can be changed as follows:

- The *open*FT administrator can change the default setting with the *ftlang* tool. Only the setting specified via the *ftlang* tool is relevant for the output of the man pages.
- Each user can change his or her own language setting using the *OPENFTLANG* environment variable. The user must enter the first two letters of the language setting in the *LANG* variable (*de* or *en*) and then export the environment variable.

  Example

  `OPENFTLANG=de; export OPENFTLANG` corresponds to (for example): `LANG=De_DE.88591,De_DE.646,`etc.

  or

  `OPENFTLANG=en; export OPENFTLANG` corresponds to (for example): `LANG=En_US.ASCII,En_US.88591,`etc.

The following table shows the effects of setting (or not setting) the OPENFTLANG and LANG variables:

OPENFTLANG	**LANG**	**Result**
Not set or empty	Not set or empty	Default setting
Not set or empty	Invalid value	Default setting
Not set or empty	Valid language	Language set in LANG
Invalid value or a language that is not installed	Not evaluated	Default setting
Valid value (2 letters, both lower case, of an installed language)	Not evaluated	Language set in OPENFTLANG

2.6 Administering requests

The request queue stores all asynchronous outbound requests, and all inbound requests. As the administrator, you can

- **obtain information** about all asynchronous requests on your system that are not yet completed. This includes the right to query information about all requests of all users. You can display the request queue with the *ftshwr* command.
- **modify** the **processing order** of all requests on your system, including those of other users. You can do this by using the *ftmodr* command.
- **cancel** asynchronous requests on your system, including those of other users. You can do this by using the *ftcanr* command.

You can also view the request queue in the graphical interface by clicking on the *Request Queue* object window. In addition, you can also execute the following functions via the graphical interface:
– Cancellation of asynchronous requests
– Update the request queue
– Change the priority of requests

You will find detailed descriptions of the functions in the online help system of the graphical interface.

2.7 Administering partners

*open*FT allows you to perform file transfers with a number of different partner systems. These partner systems may be accessible via different transport systems and protocols. *open*FT provides you with the following methods for administering these partner systems efficiently:

– The partner list

– The *Partner* object directory in the graphical user interface

– The Transport Name Service (TNS)

Partner list

You enter the address of a partner system and give it a symbolic name in the partner list. This name can be used to address the partner in all FT requests. This applies to both requests sent via the graphical user interface and requests which are issued by means of a command, via the program interface or via the *open*FT-Script interface. Although entry in the partner list is optional, it offers the following advantages:

– For each request, you may enter the short symbolic name and do not have to note the possibly complex partner address.

– You can enter routing information should the partner only be accessible via a gateway such as *open*FTIF.

– You can specify a partner instance ID which differs from the standard ID.

– You can make certain partner-specific attribute settings, e.g. the security level, the sender verification, the status (activated/deactivated), and tracing.

Partner systems with which file transfer is frequently performed should always be entered in the partner list. For more detailed information, see section “Setting up and administering the partner list” on page 57.

The Partner object directory in the *open*FT Desktop

In this directory, you enter the partner systems you want to work with as if they were network drives, i.e.:

– View directories and file attributes by clicking the mouse

– Issue file transfer requests using drag&drop

When you make such entries, you enter either the name of the partner from the partner list or the partner's address together with the transfer admission data. You can also enter a directory that is different from the home directory.

Transport Name Service

Partner systems only have to be entered in the TNS if they are not connected via the TCP/IP transport system. To use the TNS, you must explicitly activate the function in the operating parameters. To do this, you either enter the *ftmodo -tns=y* command or activate the TNS operating parameter option via the graphical user interface.

For details, see section “Entering transport system applications in the TNS” on page 192.

2.8 Authentication

If data requiring an extremely high degree of security is to be transferred, it is important that the respective partner system undergo a reliable identity check (“authentication”) before the transfer. The two *open*FT instances that are engaged in a transfer must be able to mutually check each other using cryptographic means, to ensure that they are connected to the “correct” partner instance.

In versions of *open*FT after version 8.1, for UNIX systems and Windows systems or version 9.0 for BS2000 and z/OS, an expanded addressing and authentication concept is supported. This is based on the addressing of the *open*FT instances, using a network-wide, unique ID, and the exchange of partner-specific key information.

When communicating with partners that are using *open*FT version 8.0 (or older), the functions described in the following are not usable. The previous addressing concept is still supported for these partners for the sake of compatibility. In FTAM partners, authentication is not available in this form, since the FTAM protocol standardized by the ISO does not provide for comparable functionality.

2.8.1 Instance Identifications

Each *open*FT instance that works with authentication, must be assigned a network-wide, unique instance identification (instance ID). The instance ID replaces the previous addressing of *open*FT instances using processor and application names. The instance ID is a unique name up to 64 characters long, which must not be case-sensitive. An instance ID may consist of alphanumeric characters as well as special characters. It is advisable to use only the special characters ".", "-", ":" and "%". The first character must be alphanumeric or be the special character “%”. The character “%” can only be used as an initial character. An alphanumeric character must follow a “.”.

In order to ensure the network-wide, uniqueness of the instance ID, you should proceed as follows when allocating the instance IDs:

- If the *open*FT instance has a network address with a **DNS name** you should use this as the ID. You can create an “artificial” DNS name for an openFT instance, by placing another part of a name in front of an existing “neighboring” DNS name, separated by a period.

- If the *open*FT instance does not have a DNS name, but is connected to a TCP/IP network, you should use the following ID.
  - IPv4: **%ip***n.n.n.n* (*n.n.n.n* is the IPv4 address of the local *open*FT instance without leading zeros in the address components).
  - IPv6: **%ip6[x:x:x:x:x:x:x:x]** (without scope ID) or
    IPv6: **%ip6[x:x:x:x:x:x:x:x%s]** (with Scope ID)
    where x:x:x:x:x:x:x:x is the IPv6 address of the local *open*FT instance and s is the scope ID of the local network card.

You currently allocate these IDs for your local *open*FT instances with the parameter *-id=* of the *ftmodo* command.

Instance IDs of partner systems should, from your local system's point of view, correspond to the partner name, by which the partner system is known in the *open*FT. This can be done either implicitly (name resolution via DNS/NIS, entry in the */etc/hosts*) or explicitly, by an entry in the TNS. The global name must then correspond to the instance ID of the partner. With the aid of the instance IDs of the partner systems, *open*FT administers operational resources like, for example, request waiting queues and cryptographic keys.

2.8.2 Creating and administering RSA key pairs

A suitable, public key for the given instance must be made available to the partner system, so that your own *open*FT instance can be authenticated in the partner system. Using *ftcrek* (or by via the graphical user interface), create RSA key pairs for the local *open*FT instance that currently consist of a private key and a public key. A key pair set in the UNIX system currently consists of a key pair with a length of 768 and 1024 bits. Private keys are internally administered by *open*FT, public keys are stored in the *config* directory of the instance file tree of the *open*FT instance (Standard: */var/openFT/std/config)* under the name **syspkf.r<key reference>.l<key length>**. The key reference is a numerical designator for the version of the key pair. The public key files are text files that are created using the character code of the respective operating system, i.e. EBCDIC.DF04-1 for BS2000 and z/OS, ISO8859-1 for UNIX systems and Windows.

In the *syspkf.comment* file in the *config* directory of the instance file tree, you can store comments, which are written in the first lines of the public key files when a key pair set is created. The *syspkf.comment* is a text file that you can edit. The comments could, for example, contain the contact information of the FT administrator on duty, the computer name, or similar information that is important for partners. The lines in the file *syspkf.comment* can only be a maximum of 78 characters long. Using the command *ftupdk,* you can also import subsequent comments from this file into existing public key files.

If a public key file were accidentally deleted, you could re-create the public key files of the existing key pair set using *ftupdk*.

If you want to replace a key pair set with a completely new one, you can create a new key pair set using *ftcrek.* You will recognize the most up-to-date, public key by the highest value key reference in the file name. *open*FT supports a maximum of three key pair sets at a time. The existence of several keys, however, should be temporary, until you have made the most up-to-date public key available to all partner systems. Thereafter, you can delete key pair sets that are no longer needed using *ftdelk* . Deleted key pair sets can not be restored using *ftupdk*.

2.8.3 Distributing the keys to partner systems

Distribution of public key files to your partner systems should take place using reliable means, for example by

- distributing them via cryptographically secure by e-mail
- distributing them on a CD (by courier or by registered mail).
- distributing them via a central, *open*FT file server, whose public key is in the partners' possession.

If you want to transmit your public key files to partner systems using BS2000 or z/OS or OS/390, you must ensure that these files are re-coded from ISO 8859-1 to EBCDIC.DF04-1 (e.g. by transferring them as text files via *open*FT).

The public key file of your local *open*FT instance is stored in the partner system in the following location:

- For partners using *open*FT for BS2000 as type D, PLAM elements in the library **SYSKEY** on the configuration user ID of the partner instance. The partner name allocated to your openFT instance in the remote network description file must be selected as the element name.
- For partners using *open*FT for UNIX systems in the directory **/var/openFT/*instance*/syskey**. The instance ID of your local openFT instance must be selected as the file name. The file name must not contain any uppercase characters. If the instance ID contains any uppercase characters, they must be converted to lowercase characters in the file name.
- For partners using *open*FT for Windows in the directory **openFT\var*instance*\syskey**. The instance ID of your local openFT instance must be selected as the file name.
- For partners using *open*FT for z/OS or OS/390 as a PO element in the library ***admuser*.SYSKEY.** The partner name allocated to your openFT instance in the remote network description file must be selected as the element name.

2.8.4 Administering the keys of partner systems

The public keys of the partner systems are stored in UNIX systems as files in the directory *syskey* of the instance file tree of the local *open*FT instance (Standard: */var/openFT/std/syskey*). The instance ID of the partner system must be selected as the file name. The file name must not contain any uppercase characters. If the ID contains any uppercase characters, they must be converted to lowercase characters. If an updated, public key is made available by the partner instance, the old key file must be overwritten at that time.

2.8.5 Reciprocal authentication

Basically, there are three distinct usages:

- For the local *open*FT instance, it is important that the supplied data comes from a secure source.

  To ensure this, the local *open*FT instance checks the identity of the partner instance. This assumes that a current, public key of the partner instance was stored locally in the *syskey* directory, the name of which corresponds to the instance ID of the partner instance.

  A configuration of this kind makes sense, for example, if a file server's files are to be accessed via openFT. It is important for the local *open*FT instance, that the received data come from a reliable source (the authenticated partner). In contrast, the source of an access attempt is unimportant to the file server.

- For the partner system, it is important that only a secure local *open*FT Instance is able to access its data.

  To ensure this, the partner instance checks the identity of the local *open*FT instance. This requires that a current, public key of the local *open*FT instance is stored in the partner instance (re-coded for BS2000- and z/OS- or OS/390 partners).

  A configuration of this kind would be conceivable, for example, if partner systems in several branch offices were to be accessed from a central computer via *open*FT and the branch computers were only permitted to access the central computer (and, in fact, only the central computer).

- For both the local *open*FT instance and the partner instance, it is important that the data comes from a reliable source and ends up in safe hands.

  To ensure this, both instances check the identity of the reciprocating system. For this to be possible, both public keys must have been exchanged and stored.

2.9 FT logging

As an *open*FT or FTAC administrator, you may display and delete the log records of all users.

Displaying log records

You can use the *ftshwl* command to view all log records in the system. The output of a log record contains an RC column which indicates the cause of rejection or abort of the request by means of a 4-digit reason code. This column can also contain a positive acknowledgment to a request (reason code 0000). You can use the *fthelp* command to determine the meaning of the reason codes.

Deleting log records

FT and FTAC log records may be deleted by the *open*FT administrator and the FTAC administrator. To do this, use the *ftdell* command.

Basically, *open*FT writes an indefinite number of log records. However, if no more storage space is available on disk, FT requests are rejected. If you need continuous documentation over an extended period, you should therefore back up the existing records from time to time (e.g. by redirecting the output of *ftshwl* to a printer or to disk) and then remove these log records from the current log file. The benefit of this is, first, that the log records provide a complete documentation which can be maintained over long periods, and second, that the log file does not become unnecessarily large, thus resulting in slower access performance.

Deleting log records causes the size of the log file to change since the storage space is immediately free upon deletion. On starting up the operating system, all log records older than 30 days are deleted by default.

You can also view log records in the graphical interface by clicking on the *Logging* object window. You can also execute the following functions via the graphical interface:
– Delete log records
– Select log records
– Update log window

You will find a detailed description of each of the functions in the online help system of the graphical interface.

2.10 Administering the FTAC environment

The term FTAC environment refers to the admission sets and admission profiles present on your system.

2.10.1 Administering admission sets

As the FTAC administrator, you specify the standard admission set and can view, modify and delete the standard admission sets for all users in the system.

Standard admission set

The standard admission set applies to all login names. The user can restrict this admission set further.

The user can override the entries in the standard admission set only,
– if you, as FTAC administrator, modify the admission set of the user accordingly,
– or if you set up a privileged FT profile.

Following an initial installation or preinstallation of *open*FT, the standard admission set is set so that file transfer is possible without restriction. As FTAC administrator, you should therefore adapt the standard admission set to the protection requirements on your processor.

Displaying and modifying admission sets

Admission sets can be viewed using the *ftshwa* command. The entries made by the FTAC administrator are listed under MAX-ADM-LEVELS, the user entries under MAX-USER-LEVELS. The smaller value is valid in each case.

You can also view admission sets in the graphical interface by clicking on the *Admission Sets* object window. You will find a detailed description of each of the functions in the online help system of the graphical interface.

The settings in the admission set apply to all users initially. As the FTAC administrator, you can assign an individual admission set for each user in the system or modify an existing one. The *ftmoda* command is available for this purpose.

Using admission sets properly

To protect your processor against serious attempted intrusion, you should set the inbound properties in the admission set as restrictively as possible for the login name *root*, i.e. at least prohibit inbound processing.

1. For secure operation, you should prevent all inbound admissions in the standard admission set, e.g. by using the command:

   ```
   ftmoda @s -os=100 -or=100 -is=0 -ir=0 -if=0 -ip=0
   ```

2. For each user authorized to carry out inbound processing, you, as FTAC administrator, should set all parameters of the corresponding admission set to 100.

3. Recommend all users to change their inbound values to 0. They may then use their profiles and the “ignore ... level” function to permit any desired access mode. File transfers will then be allowed only via the FTAC transfer admission, but no longer via the login and password.

It is also possible,

– to assign partner-specific security levels

– and for *open*FT partner to undergo a reliable identity check using cryptographic means, see section “Authentication” on page 27.

The use of a file name prefix in the FT profile provides additional security. This prevents switching to a parent directory.

Important

If you have high security requirements, these actions are really only useful if *open*FT is the only active application for file transfer tasks on your processor, i.e. TCP/IP services like *ftp*, *tftp* must not be active!

2.10.2 Administering admission profiles

As the FTAC administrator, you can create FT profiles for any user in the system and modify them later. The FTAC administrator is the only person who can assign privileges to FT profiles.

Creating FT profiles

You can create FT profiles with the command *ftcrep*. If you also want to assign a transfer admission at the same time, you must either have *root* authorization as the FTAC administrator or specify the password for the particular login name. If you do not have *root* authorization or specify the password, the profile is created without a transfer admission; the user must then assign it later.

When you create the profile, you can also assign privileges.

You can also create admission profiles in the graphical interface by opening the *Admission Profiles* dialog window via the *File/New* menu item. You will find a detailed description of each of the functions in the online help system of the graphical interface.

Viewing and modifying FT profiles

You can use the *ftshwp* command to display the FT profiles of all users. The transfer admission of the profile is not output, i.e. your administrator privileges do not grant you access to files on remote systems.

You can also view the admission profiles in the graphical interface by clicking on the *Admission Profiles* object window. You can also change admission profiles in the *Admission Profiles* dialog window. You will find a detailed description of each of the functions in the online help system of the graphical interface.

You can use the *ftmodp* command to make the following changes to an FT profile:
– assign or cancel privileges
– modify the transfer admission, if you have *root* authorization or know the password
– assign the profile to another login name

Following a modification of this nature, the profile will be locked, unless the FTAC administrator *root* has authorization (UID=0), and must be explicitly unlocked, e.g. by using the command *ftmodp ... -v=y*.

If a transfer admission is assigned for a second time, the existing transfer admission is locked.

Deleting FT profiles

You can use the *ftdelp* command to delete FT profiles of a user. This function is necessary, for example, after deletion of a login name, since the profiles are not automatically deleted when a login name is deleted. You should contact the user before you delete profiles from active login names.

You can also delete admission profiles via the graphical interface by selecting the *Delete* command from the context menu. You will find a detailed description of the object windows in the online help system of the graphical interface.

Assigning privileges to FT profiles

A privileged FT profile is intended for exceptional circumstances in which it is necessary for a user to override all restrictions. To assign privileges to a profile, you can use the command *ftmodp ... -priv=y*, for example.

Once a profile has been assigned privileges, it is possible only to modify the transfer admission and cancel the privileges. To prevent abuse, no other changes are permitted.

You can also assign privileges to admission profiles via the graphical interface in the *Admission Profiles* dialog window. You will find a detailed description of each of the functions in the online help system of the graphical interface.

2.10.3 Saving the FTAC environment

When migrating individual users to another processor, or when migrating the complete processor, it is possible to provide the users with the same FTAC environment by saving the admission sets and FT profiles and restoring them on the new processor. Furthermore, you can also created backup copies of the FTAC environment on your processor by this method.

Saving admission sets and FT profiles

You can use the *ftexpe* command for backups. You can select the admission sets and FT profiles which you wish to save for particular users. You must specify the name of the backup file.

In all cases, the standard admission set is not included in the backup. Instead, all the values of an admission set that refer to the standard admission set (represented by an asterisk (*) in the display) are stored as variables. This means that when they are restored, they will receive the value of the standard admission set valid at the time.

You can also save admission sets and admission profiles via the graphical interface using the *Export FTAC Environment* command in the *Administration* menu. You will find a detailed description of each of the functions in the online help system of the graphical interface.

Displaying saved admission sets and FT profiles

You can display saved admission sets and FT profiles with the *ftshwe* command. You must specify the name of the backup file.

You can also view saved admission sets and admission profiles via the graphical interface by dragging the export file into the *Exported Admissions* directory and then dropping it there.

Importing saved admission sets and FT profiles

You can re-import saved admission sets and FT profiles with the *ftimpe* command. Here, you must make a distinction between sets, profiles and login names, i.e. you must not accept the entire backup contents. Please note that the values which refer to the standard admission set are always assigned the values of the currently valid admission set.

If you have *root* authorization as the FTAC administrator, the admission profiles that you import will be immediately available with the status that was set on exporting the profile. If you do not have *root* authorization, imported profiles will initially remain locked for the login names (or user IDs) of other users.

You can also import admission sets and admission profiles via the graphical interface using the *Import FTAC Environment* command in the *Administration* menu. You will find a detailed description of each of the functions in the online help system of the graphical interface.

2.11 Using *open*FT in a cluster

With *open*FT, you can run several *open*FT instances at the same time on a single host. These instances allow you to switch to a different computer already running *open*FT so that you can continue to use the *open*FT functionality when the initial host fails. You will find examples on how to use *open*FT in a cluster of UNIX systems in the appendix.

A requirement for this is that *open*FT uses only the TCP/IP transport system. Other transport systems are not supported in a cluster and must also not be configured in the TNS. In a cluster, the same version of *open*FT must be running on all the computers.

For systems that do not have TCP/IP there is currently only the standard instance.

*Open*FT commands that call preprocessing, postprocessing or follow-up processing run in the same instance as the request that initiated the pre-, post- or follow-up processing.

If you administer *open*FT via SNMP, then please note when switching to the cluster that SNMP can only work together with one instance.
The decisive factor is which instance is set when the agent is started (see also chapter “Administering openFT via SNMP” on page 59).

Command for administering instances

As an *open*FT administrator you can create, modify and delete instances. You can also set up instances and obtain information on instances (like a user).

- Creating or activating an instance

  Using the command *ftcrei,* you can create a new instance or re-activate (switch on) a deactivated instance.

  When an instance is created, the operating parameters, the profile files, the startup and shutdown files are initialized as during a new installation.

  When an existing instance is deactivated, the existing instance file tree, with the operational resources of the instance, is linked to the directory */var/openFT.*

- Modifying an instance

  You can assign a different Internet host name to an instance with the *ftmodi* command.

- Deleting an instance

  You can delete an instance with the *ftdeli* command. Deleting an instance in this manner only removes the symbolic link in the local */var/openFT* directory. The instance file tree is not changed.

- Setting up an instance

  You can select the *open*FT instance you want to work with using the *ftseti* command.

  The command sets the OPENFTINSTANCE environment variable to the name of the instance.

  You can also set up the instance via the graphical interface. If there is more than one instance, then a list appears in the graphical interface from which you select the instance.

- Outputting information on instances

  You can query information on the instances using the *ftshwi* command.

- Updating an instance file tree

  Using the *ftupdi* command, you can modify the instance file tree of an older version of *open*FT for use in the current version. That is only necessary for instances that were not active at the time of an update installation.

– If you work with more than one instance, then in this case a separate *ftalarm* call is required for each instance (see also section “ftalarm - Report failed requests” on page 79).

– You will find detailed descriptions of the *ftcrei*, *ftmodi, ftupdi* and *ftdeli* commands in chapter “openFT commands for the administrator” starting on page 65. The *ftseti* and *ftshwi* commands are described in the “*open*FT for UNIX systems” User Guide.

Startup and shutdown file

In *open*FT, there is one global startup and shutdown file that operates on all instances. In addition, every instance present also has its own startup and shutdown file.

During a system startup / shutdown, the global startup and shutdown file is called. This file then calls the startup and shutdown files of all *open*FT instances.

- Global startup and shutdown file:

  It is set up just like the previous startup and shutdown file under */etc/init.d* or in a corresponding directory on an external platform during the installation of *open*FT. This startup and shutdown file calls the startup and shutdown files of all instances when the system is started or when it is shut down.

- Startup and shutdown file specific to one instance:

  The startup and shutdown file *openFTinst* is created in the */var/openFT/std/etcinit* directory for the *std* instance during the installation of *open*FT.

  If you create another instance with *frcrei*, then a startup and shutdown file *openFTinst* is also set up for this instance in the directory */var/openFT/instance/etcinit* (*instance* = name of the new instance).

2.12 Diagnosis

To support error diagnostics, you can switch a trace on or off, trace files and output diagnostic information. These functions are primarily intended for the Maintenance and Diagnostic Service of Fujitsu Siemens Computers.

Switching on and off trace mode

You can switch the trance mode on or off with the FT command *ftmodo* or via the graphical interface. When the trace mode is enabled, the diagnostic data is written to trace files, which must be edited for further diagnostics.

Preparing trace files

The trace files are located in the directory */var/openFT/instance/traces* where *instance* is the name of the corresponding instance. These files must be edited with the *fttrace* command.

To create a trace file You can switch the trace function on and off in the graphical interface in the *Operating Parameters* dialog window in the *Administration* menu. The trace file can be displayed using the *Open Trace File* command in the *Administration* menu. You will find a detailed description of each of the functions in the online help system of the graphical interface.

Displaying diagnostic information

Unlike trace files, diagnostic records are written only if an error occurs. You can output these diagnostic records with the *ftshwd* command.

Message file for console commands:

In order to use the diagnostic trace information in console output, the output is also stored in the file */var/openFT/instance/log/conslog,* where *instance* is the name of the corresponding instance.

3 Installation and configuration

This chapter describes the installation and configuration of *open*FT.

*open*FT is shipped with a communications manager. In the following, this communications manager is always referred to as CMX (Communications Manager for UNIX systems) even if different package names are used for the various platforms (such as CMX, PCMX, CMX.all, SMAWcmx, SMAWpcmx).

3.1 Installation of *open*FT

The installation of *open*FT is performed under the login name *root*.

The installation technique of *open*FT depends on the operating system and is described in the respective Release Notice. There are three different types of installation depending on whether an FT version is already installed or which FT version is already installed on your computer:

- Initial or full installation
  This means that your computer has an *open*FT < V8.0 or no FT version on it.
- Update installation
  This means that your computer has *open*FT version 8.0 or V8.1 installed.
- Installation of a correction version
  This means that your computer has *open*FT version 10.0 installed.

What you need to observe before installing *open*FT ...

- If CMX has not yet been installed, you will need to first install CMX from the supplied storage medium before installing *open*FT. Make sure after installing CMX that CMX is started, i.e. that the *tnsxd* process is running. This must be running before you install *open*FT.
- The language used in *open*FT (German, English) is not queried anymore during the installation. The language is now selected using the *LANG* environment variable. For this reason, the response file only contains the *FTAM* and *FTP* variable and does not contain the *LANM* variable anymore (see also section “Switching the language interface” on page 23).

- If you want to encrypt user data, you must also install *open*FT-CR for UNIX systems. This software is offered without a license at a fixed price. If an openFT-CR version < V8.0 is already installed, then you must first uninstall this version before installing *open*FT, and then you can install *open*FT-CR V10.0 .
- If you want to use the *open*FT-Script interface then the international version of the Java Runtime System, version 1.4.2_07 (or higher) must be installed. The binary directory containing the *java* executable should be present under one of the following paths:

  ```
  /opt/*/bin
  /opt/*/*/bin
  /usr/*/bin
  /usr/*/*/bin or
  /etc/alternatives/bin
  ```

  The *open*FT installation procedure then creates the reference to the Java executable which is required in UNIX systems in the *open*FT directory.

  In other cases, the installation procedure issues a warning informing you that Java could not be found. In such a case, you must install Java in one of the above-named directories and create the link to it. To do this, enter the following command:

  ```
  ftsetjava @s
  ```

  The `ftsetjava` command also allows you to check whether Java is installed and, if so, in which variant (`ftsetjava @a`) or check which Java variant is used (`ftsetjava` without parameters).

The following three sections describe which steps must be performed for the three installation variants by you as the system administrator as well as those which are handled automatically by the installation procedure.

3.1.1 Initial or full installation

If you have not installed *open*FT on your system yet, the installation is an initial installation.

If *open*FT version 7.0 (or earlier) is installed, then it is a full installation.

Tasks required of the system administrator

1. If *open*FT version 7.0 (or earlier) and possibly add-on products are already installed, then you should proceed as follows:
   - Save admission profiles and admission sets that are still needed in an external file using *ftexpe*.
   - Uninstall *open*FT and the add-on products.
2. Install the *open*FT V10 product software.

   When doing this, please note the following:
   - If you want to install *open*FT-FTAM or on a system in which the *open*FT installation takes place in a dialog, then you need to answer a question during installation asking you if you have a valid *open*FT-FTAM license and a valid *open*FT-FTP license. If answered with *yes*, then *open*FT-FTAM and/or *open*FT-FTP are installed, otherwise they are not installed.
   - This question is not asked on HP, AIX and Linux systems. If you want to use the FTAM or FTP functionality on these systems, then you must activate *open*FT-FTAM and/or *open*FT-FTP via the *install.ftam* or *install.ftp* command after installing *open*FT (see also section “install.ftam - Install openFT-FTAM” on page 164 and section “install.ftp - Install openFT-FTP” on page 165).
3. Import the saved admission sets and admission profiles using *ftimpe*. All security levels in the admission sets that were previously set at 1 are automatically converted to 90. The standard admission set is re-set.

After these steps, *open*FT will be fully operational and will be activated at each system startup.

Steps performed automatically

During installation, the following steps are carried out automatically:

- The use of the TNS is deactivated.
  Despite this, for an initial installation, standard TNS entries are created for *open*FT; for a full installation, existing entries for *open*FT are modified (see the section “TNS entries created automatically” on page 194).
- The operating parameters (e.g. maximum number of requests that can be processed simultaneously, maximum block length, scope of FT and FTAC logging, setting of the CCS) are set to default values. The node name of the processor is entered as the processor name (corresponds to the output in *uname -n*). The DNS name of the computer (if one exists) is pre-set as the instance ID for the standard instance. When there is no DNS name, the node name of the computer is used for the instance ID.
- The following startup and shutdown files are set up:
  – The startup and shutdown file that applies to all instances (e.g. */etc/init.d/openFT* under Solaris)
  – The startup and shutdown file for the *std* instance (path: */var/openFT/std/etcinit/openFTinst*).

  With the help of this file *open*FT is started automatically each time the system is started, and is terminated automatically each time the system is shut down (see also section “Using openFT in a cluster” on page 39).
- A standard admission set permitting all file transfer functions is created.
- A key pair set is created (see page 29).
- The file transfer is started (but not on HP systems).
- The system searches for a suitable Java executable and this is notified to *open*FT. If no such system is found then you must proceed as described on page 44.

3.1.2 Update installation from *open*FT V8.0 and V8.1

If *open*FT V8.0 or V8.1 is already installed, an update installation is performed.

The following points must be observed:

- The log file is deleted for an update installation from V8.0 It should be evaluated before performing the installation if necessary.
- Existing requests are deleted from the request queue unconditionally. If any follow-up processing was specified with the option *-lf=* in the submitted request, this is completed in the process.
- Existing trace files, if any, and the *DIAGFILE* are deleted.

Tasks required of the system administrator

1. Install *open*FT from the data medium.
2. If you want to install *open*FT-FTAM on a system in which the *open*FT installation takes place in a dialog, then you need to answer questions asking you if you have a valid *open*FT-FTAM license and a valid *open*FT-FTP license. Depending on the answers *open*FT-FTAM and/or *open*FT-FTP may or may not be installed.

   These questions are not asked on HP, AIX and Linux systems. If you want to use the FTAM or FTP functionality on these systems, then you must activate *open*FT-FTAM and/or *open*FT-FTP via the *install.ftam* or *install.ftp* command after installing *open*FT (see also section “install.ftam - Install openFT-FTAM” on page 164 and section “install.ftp - Install openFT-FTP” on page 165).
3. If you have made modifications in the old startup and shutdown files, in the case of an *open*FT V8.0 update installation you must also make them in the new start up and shutdown files, if applicable. See the section “Using openFT in a cluster” on page 39
4. If you want to continue to use instance directories which were deactivated with *ftdeli* prior to the update then you must update these with *ftupdi*.

Steps performed automatically

The following steps are performed automatically for an update installation:

- Current *open*FT processes and the graphical user interface are terminated.
- The TNS entries from the previous version are modified and TNS use remains activated.

- The language setting is carried over from the previous version.
- The new instance-overlapping startup and shutdown file (e.g. */etc/init.d/openFT* on Solaris) is installed. The old instance-overlapping startup and shutdown file is no longer automatically saved.
- The instance directories of currently existing instances and of the standard instance are updated. During this, the following steps are carried out:
  - Operating parameters:
    The operating parameters (e.g. maximum number of requests that are being simultaneously processed, the maximum block lengths, the scope of the FT and FTAC logging, setting the CCS and processor name, etc.) are carried over from the previous version for all *open*FT instances.

    In the case of an *open*FT V8.0 update installation the DNS name of the computer (if one exists) is preset as the instance ID for the standard instance. When there is no DNS name, the node name of the computer is used for the instance ID (corresponding to the output from *uname -n*).

    In the case of an *open*FT V8.1 update installation the instance ID of the previous version is carried over.
  - Instance-specific startup and shutdown files:
    In the case of an *open*FT V8.0 update installation, the old instance-specific startup and shutdown files */var/openFT/<instance>/etcinit/openFTinst* are stored to */var/openFT/<instance>/etcinit/openFTinst .old*.
    Subsequent to this, the new instance-specific startup and shutdown files are installed.
    In the case of an *open*FT V8.1 update installation, the instance-specific startup and shutdown files are carried over from the previous version.
  - The FTAM catalog is carried over from the previous version.
  - Logging records:
    In the case of an *open*FT V8.0 update installation the log file is deleted.
    In the case of an *open*FT V8.1 update installation, the log records are carried over from the previous version.
  - Admission sets and profiles:
    The admissions set and admissions profile are carried over from the previous version. In the case of an *open*FT V8.0 update installation, all security levels that were previously set to 1 in the admissions sets are automatically converted to 90.

  - Key pair sets:
    In the case of an *open*FT V8.0 update installation, a key pair set is created (see page 29).
    In the case of an *open*FT V8.1 update installation, all the key pair sets are carried over.

- The file transfer is started for those instances, for which it was started before the installation (not applicable on HP systems).
- The system searches for a suitable Java executable and this is notified to *open*FT. If no such system is found then you must proceed as described on page 44.

3.1.3 Installation of a patch

Installation of a patch means that *open*FT V10.0 is already installed on your computer.

Tasks required of the system administrator

1. Install *open*FT V10.0 from the data medium.
2. If you want to install *open*FT-FTAM on a system in which the *open*FT installation takes place in a dialog, then you need to answer questions asking you if you have a valid *open*FT-FTAM license and a valid *open*FT-FTP license. If answered with *yes*, then *open*FT-FTAM and/or *open*FT-FTP are installed, otherwise they are not installed.

   This question is not asked on HP, AIX and Linux systems. *open*FT-FTAM and *open*FT-FTP are automatically installed on these systems if they were installed in the previous version.

Steps performed automatically

The following steps are performed automatically on installing a patch:

- Current *open*FT processes and graphical user interfaces are terminated.
- The FT profiles and admission sets, the log files, the startup and shutdown files, the FTAM catalog, operating parameters and requests, the partner list, and the key pair sets are taken over without changes for all *open*FT instances.
- If you work on an HP, AIX or Linux system, then *open*FT-FTAM and *open*FT-FTP are automatically installed on these systems if they were installed in the previous version.
- The language setting from the previous version is used.
- The file transfer is started for those instances, for which it was started before the installation (not applicable on HP systems).

3.1.4 Activities after installation

Following the installation of *open*FT, you may need to perform additional steps, depending on what you require of your system. These may include the following:

- encryption
- distributing public keys and obtaining public keys for partner systems needing to be authenticated.
- Identifying instances and specifying the name of the local system for *open*FT
- disabling automatic startup of *open*FT
- automatic saving of log records in files, followed by deletion
- activating *ftalarm* function
- starting *open*FT subagents automatically
- installing and uninstalling *open*FT-FTAM
- installing and uninstalling *open*FT-FTP
- only under Linux: setting up authentication via PAM

If you still use the TNS then you may need to create the TNS entries, see section “Entering transport system applications in the TNS” on page 192.

Please note that cluster configurations are only supported for TCP/IP. You will therefore need to check all *open*FT-specific TNS entries for cluster configurations and delete those transport system entries that are not related to TCP/IP. (i.e. everything but RFC1006 and LANINET).

Encryption

If you want to use encryption for user data in addition to request description data, you must install *open*FT-CR version 10.0 for UNIX systems.

When connecting to *open*FT partners that support the AES algorithm, the request description data and user data are encrypted using the new RSA/AES algorithm (instead of with the previous RSA/DES algorithm).

So that you can transfer *open*FT request description data and file content in encrypted form, there must be a key pair set in the local system (see page 29). A key pair set is created during installation of *open*FT and consists of private and public keys of suitable length.

Other key pair sets can be created (if necessary) using *ftcrek*. Obsolete key pair sets are deleted using *ftdelk*.

Private keys are internally administered by *open*FT. Public keys are saved under the name **syspkf.r<key reference>.l<key length>** in the *config* directory of the instance file tree of the *open*FT instance (standard: */var/openFT/std/config*). The key reference is a numerical designator for the key pair version.

Distributing public keys and obtaining public keys for partner systems to be authenticated

If your local system is to be authenticated in partner systems, then public keys for your system need to be made available to the partner systems. You can find details in the section “Distributing the keys to partner systems” on page 30.

If partner systems are to be authenticated by *open*FT, you will need the public keys of those partners. The public keys of the partner system are stored in the UNIX system as files in the directory *syskey* of the instance file tree of the local *open*FT instance (standard: */var/openFT/std/syskey*). The instance ID of the partner system must be selected as the file name. The data name must not contain any uppercase characters. If the ID contains uppercase characters, these must be converted to lowercase characters in the file name. If an updated public key is made available by the partner instance, the old key file must be overwritten.

Specifying the instance ID and the name of the local system for *open*FT

*open*FT sends a sender address along with the request to a remote system. This sender address must be known to *open*FT before you issue requests. Partner systems using *open*FT version 8.1 and later, are identified by the so-called “instance ID.” The local instance ID is defined using the command *fmodo -id=* (or by using the graphical user interface). You will find details on this in the section “Instance Identifications” on page 27.

For connecting to an older version of *open*FT on BS2000/OSD, OS/390 or z/OS, *open*FT needs a sender address. With a processor link, the node name of your processor is also sent as the sender address. The network administrator for your processor has stipulated the node name for your processor (*uname -n*). With installation of *open*FT, the node name is automatically entered as the processor name. In this case, you do not have to take any action.

More details on the *ftmodo* command and the *-id, -p* and *-l* options can be found in the description on the *ftmodo* command starting on page 106.

Disabling the automatic startup of *open*FT

During installation, the startup file is installed (e.g. */etc/rc2.d/S73openFT* on Solaris), with which file transfer is automatically initiated at system startup. This script calls the file */var/openFT/std/etcinit/openFTinst* when the system starts, which then automatically starts the file transfer.

If *open*FT instances were created using the *ftcrei* command, then this script also calls the startup and shutdown file for this instance (see also section “Using openFT in a cluster” on page 39).

These files then start the file transfer for the corresponding instance.

If you do not want file transfer to be started automatically, you will need to comment out the corresponding command line in the file */var/openFT/std/etcinit/openFTinst* or in the startup and shutdown file for the instances.

Automatic termination of *open*FT

During installation, the shutdown file is installed (e.g. */etc/rc0.d/K10openFT* on Solaris). This script calls the file */var/openFT/std/etcinit/openFTinst* when the system shuts down, which then automatically terminates the file transfer.

If *open*FT instances were created using the *ftcrei* command, then this script also calls the startup and shutdown file for this instance (see also section “Using openFT in a cluster” on page 39).

These files then terminate the file transfer for the corresponding instance.

If you do not want file transfer to be terminated automatically, you will need to comment out the corresponding command line in the file */var/openFT/std/etcinit/openFTinst* or in the startup and shutdown file for the instances.

Automatic saving of log records in files, followed by deletion

The logging file can grow exponentially and quickly fill the disk on which it is saved. It is therefore extremely important that this file be monitored and purged on a regular basis.

In order to keep the size of the logging file as small as possible, all log records older than 30 days are automatically deleted whenever the system is started. If you want to define some other time period, you will need to modify the corresponding line in the startup and shutdown file */var/openFT/std/etcinit/openFTinst* and/or in the startup and shutdown files of other instances.

If you also want all log records to be to be saved before being deleted, you can add an appropriate *ftshwl* command in the startup and shutdown file. An example of this is already included as a comment in the startup file.
You will find an example for the cyclical deletion of log records at *http://www.fujitsu-siemens.com/openft* under the item *Application Scenarios.*

Enabling the ftalarm command

If you want to be informed about the frequency of failed FT requests, it is advisable to use the *ftalarm* command for this purpose. If desired, you can also have the *ftalarm* command automatically started at system startup by inserting a corresponding line with the *ftalarm* command in the startup and shutdown file */var/openFT/std/etcinit/openFTinst* and/or in the startup and shutdown files of other instances.

Starting the *open*FT subagent automatically

If you want to automatically start the *open*FT subagent for administration using SNMP at system startup, you must activate the corresponding line with the *ftagt* command in the startup and shutdown file */var/openFT/std/etcinit/openFTinst* and/or in the startup and shutdown files of other instances.

More details on this can be found in the chapter “Administering openFT via SNMP” on page 59.

Please note for clusters that SNMP can only work with a single instance. The deciding factor is which instance is set up when the agent is started (see also section “Using openFT in a cluster” on page 39).

Installing and uninstalling *open*FT-FTAM

*open*FT-FTAM is not installed together with *open*FT when the installation is an initial or full or update installation on an HP, AIX or Linux system. The same applies to patch installations when *open*FT-FTAM was not installed beforehand.

In these cases you need to install *open*FT-FTAM using the *install.ftam* command after installing *open*FT. You will find this command in the directory */opt/openFT/bin/ftbin*, see also section “install.ftam - Install openFT-FTAM” on page 164.

Installation is only permitted when you also have a valid *open*FT-FTAM license available.

You can also uninstall *open*FT-FTAM if it is not needed anymore using *install.ftam*. *open*FT-FTAM must be uninstalled if you do not have the corresponding license.

Installing and uninstalling *open*FT-FTP

*open*FT-FTP is not installed together with *open*FT when the installation is an initial or full or update installation on an HP, AIX or Linux system. The same applies to patch installations when *open*FT-FTP was not installed beforehand.

In these cases you need to install *open*FT-FTP using the *install.ftp* command after installing *open*FT. You will find this command in the directory */opt/openFT/bin/ftbin*, see also page 165.

Installation is only permitted when you also have a valid *open*FT-FTP license available.

You can also uninstall *open*FT-FTP if it is not needed anymore using *install.ftp*. *open*FT-FTP must be uninstalled if you do not have the corresponding license.

Authentication via PAM under Linux

As of *open*FT version 10, the PAM interface for user authentication is supported under the Linux operating system. PAM (Pluggable Authentication Modules) consists of a collection of program libraries which allow system administrators to choose the way applications authenticate users. This is controlled by means of application-specific configuration files in the directory */etc/pam.d* or by means of an entry in the file */etc/pam.conf* if */etc/pam.d* does not exist.

When logging on to PAM, *open*FT uses the service name *openft*. In the case of an *open*FT update/initial installation, a configuration file with the name *openft* is therefore created in the directory */etc/pam.d* if no such file already exists. The

authentication mechanism that is to be used is defined in this file. If the system administrator has defined a specific authentication mechanism via the file */etc/pam.d/common-auth* then this is used by *open*FT. If not, the PAM module *pam_unix.so* for user authentication under Linux is used.

If the directory */etc/pam.d* does not exist then the system administrator must make a suitable entry in the file */etc/pam.conf* for the service name *openft*.

3.1.5 Automatic installation

You may also select automatic installation when installing *open*FT on some systems. In this case, installation is carried out without user prompts on screen. The additional data required for installation of *open*FT-FTAM and *open*FT-FTP are taken from the *response* file. For *open*FT, a standard response file is supplied. It has the following contents:

```
FTAM='NO'
FTP='NO'
```

Meaning of the environment variable

FTAM

specifies whether or not you are authorized to use the FTAM functionality, i.e. whether or not you have an *open*FT-FTAM license. In the standard response file, this variable is preset to *NO*, i.e. *open*FT-FTAM is not installed.

Other possible values:
YES, i.e. you are authorized to use the FTAM functionality, i.e. you have an *open*FT-FTAM license. *open*FT-FTAM is therefore installed. You are then able to use *open*FT-FTAM.

FTP

specifies whether or not you are authorized to use the FTP functionality, i.e. whether or not you have an *open*FT-FTP license. In the standard response file, this variable is preset to *NO*, i.e. *open*FT-FTP is not installed.

Other possible values:

YES, i.e. you are authorized to use the FTP functionality, i.e. you have an *open*FT-FTP license. *open*FT-FTP is therefore installed. You are then able to use *open*FT-FTP.

3.2 Setting up and administering the partner list

Although the creation of a partner list is optional, it offers considerable advantages. These include simplified addressing for users, the central administration of partner addresses and enhanced security since you can assign individual properties such as security level or partner check level to partner systems.

Following a new installation, the partner list is empty. Consequently, you should create the partner list immediately after installation and, in particular, enter frequently used partners in this list.

You can use the following commands to administer the partner list:

– *ftaddptn*: Enter new partner in the partner list

– *ftmodptn*: Modify the properties of a partner in the partner list

– *ftremptn*: Remove a partner from the partner list

– *ftshwptn*: Display the properties of partners in the partner list and export the partner list

You can also administer the partner list via the graphical user interface:

– Menu command *File - New - Partner List Entry* ... : Enter new partner in the partner list

– *Partner List* object window: You can enter new partners, remove partners and modify partner properties via the context menu.

For further details, refer to the online help system.

Dynamic partners

Users may, as required, specify partners via the name in the partner list or via their addresses (dynamic partners). In this way, they can also address partners that are not entered in the partner list.

As FT administrator, you may also lock the partner list for security reasons. To do this, use the *ftmodo -dp* command or select *Administration - Operational Parameters* from the menu. In this case, it is necessary to address partners via their names in the partner list. They cannot be addressed directly via their address.

Exporting the partner list

You can use the *ftshwptn* command to export the partner list entries to a file, for example in order to back up the entries or use them in other systems. On export, the entries are converted into the corresponding commands (*ftmodptn*) which you simply need to read in.

In *ftshwptn* you also specify the platform for which the commands are to be generated.

Examples

– To back up the partner list in a format for UNIX systems in the file *ftpartner.sav*:

```
ftshwptn -px > ftpartner.sav
```

You can re-import the partner list by calling the file as a procedure file, e.g. with

```
sh ftpartner.sav
```

– To export the partner list in BS2000 format to the file *ftpartner.bs2*:

```
ftshwptn -p2 > ftpartner.bs2
```

4 Administering *open*FT via SNMP

In order to administrate *open*FT via SNMP, your processor must be have a EMANATE master agent.

The *open*FT subagent is available for Solaris/Sparc and HP-UX platforms. It is supplied with *open*FT and is set up when *open*FT is installed.

4.1 Activities after installation

After installation of *open*FT, different activities are required.

1. If your system is not already being administered with SNMP, you will need to activate administration via SNMP.

   You will need a community string with write authorization to administer *open*FT via the *open*FT subagent. If you only have read authorization, then only information can be output via SNMP. In this case you will not be able to change values (or perform starts or stops, see also page 61).

   Consult your plattform specific documentation to find out how to activate the SNMP administration.

2. Start the agent (see below)

You will find a list of activities performed by the SNMP administrator in the documentation for the management station used.

Consult your SNMP documentation to obtain information on security mechanisms.

4.2 Starting the *open*FT subagent

There are two ways to start the *open*FT subagent:

- Enter `/opt/bin/ftagt &`.

  The *open*FT subagent is then started and remains active until the system is shutdown.

- Remove the comment symbol in the line of the startup file that contains the word *ftagt* (for example: */var/openFT/std/etcinit/openFTinst*) as well as in the corresponding line in the startup file of any other instances. The *open*FT subagent is then also started each time the system is booted.
  If you want to terminate the *open*FT subagents for some reason, then you can do this with a *kill -2* command with the process number of the *open*FT subagent as the parameter.

Note that SNMP can only work with one instance when clustered. The decisive factor is which instance is set up to start when the agent is started (see also section “Using openFT in a cluster” on page 39).

4.3 SNMP management for *open*FT

The *open*FT subagent is used to:

- obtain information about the status of *open*FT
- start and stop *open*FT
- obtain information about system parameters
- modify system parameters
- create the new public key for encryption
- output statistical data
- to control the diagnosis

The MIB to *open*FT offers objects for the above-mentioned management tasks. It is located in the file */opt/openFT/snmp/openFT.asn1*.

The objects for starting and stopping, encrypting the public key, modifying the system parameters and controlling the diagnose require write access.

4.3.1 Starting and stopping *open*FT

MIB definition

Object name	Access	TransView interface
ftStartandStop ftStartandStopFTAM	read-write read-only	*open*FT protocol FTAM protocol

Entry

Syntax	Integer	Meaning
on off	3 4	*open*FT/FTAM is started *open*FT/FTAM is stopped

Setting the values “on” or “off” causes the *open*FT subagent to start or stop *open*FT. Write access supplies information about the current status of the FT system.

4.3.2 System parameters

MIB definition

Object name	Access	TransView interface
ftSysparVersion	read-only	Version
ftSysparTransportUnitSize	read-write	Transport Unit Size
ftSysparMaxOSP	read-write	Max OSP
ftSysparMaxISP	read-write	Max ISP
ftSysparProcessorName	read-write	Processor Name
ftSysparStationName	read-write	Station Name
ftSysparCode	read-write	Code Table
ftSysparMaxInboundReqs	read-write	Max Inbound Requests
ftSysparMaxLifeTime	read-write	Max Life Time

The explanation of the possible values in the description of the *ftmodo* command starting on page 106.

4.3.3 Statistical information

MIB definition

Object name	Access	TransView interface
ftStatLocked	read-only	Requests in status LOCKED
ftStatWait	read-only	Requests in status WAIT
ftStatActive	read-only	Requests in status ACTIVE
ftStatCancelled	read-only	Requests in status CANCELLED
ftStatFinished	read-only	Requests in status FINISHED
ftStatHold	read-only	Requests in status HOLD
ftStatLocalReqs	read-only	Local requests
ftStatRemoteReqs	read-only	Remote requests

The individual states have the following meanings:

LOCKED
The request is temporarily excluded from processing.
This state may occur both for *open*FT and for FTAM partners.
With *open*FT partners, e.g. when a resource bottleneck is encountered or when external data media must be made available.
With FTAM partners, when one of the partners proposes a waiting period until the next start or recovery attempt via the FTAM protocol, and this period exceeds the delay normally permitted.

WAIT
The request is waiting.

ACTIVE
The request is currently being processed.

CANCELLED
The request was cancelled in the local system. However, the remote system is aware of its existence, e.g. because it was previously active. Therefore, the request cannot be removed from the request queue until a connection to the partner has been re-established.

FINISHED

This status arises for requests involving FTAM partners when the request has been either completed or cancelled, but the user has not yet been informed of the fact

HOLD

The start time specified when the request was issued has not been reached

4.3.4 Control of diagnostics

MIB definition

Object name	Access	TransView interface
ftDiagStatus	read-write	Diagnose Management

Entry

Syntax	Integer	Meaning
off on	1 18	Diagnosis management is deactivated Diagnosis management is activated

4.3.5 Public key for encryption

MIB definition

Object name	Access	TransView interface
ftEncryptKey	write-only	

Entry

Syntax	Integer	Meaning
create-new-key	1	A new public key is created.

5 *open*FT commands for the administrator

This chapter contains the commands which are available only to the administrator or which include more options for the administrator than the user.

The commands for the *open*FT script interface are described in the User Guide as well as in the *open*FT Script Interface manual.

5.1 Overview of the commands

Command	Function	Note
ftaddptn	Enter partner in the partner list	FT administrator only
ftalarm	Report failed requests	FT administrator only
ftcanr	Delete asynchronous requests	FT user also [1) 2)]
ftcrei	Create an instance	FT administrator only
ftcrek	Create a key pair set	FT administrator only
ftcrep	Create FT profile	FT user also [1) 3)]
ftdeli	Delete an instance	FT administrator only
ftdelk	Delete a key pair set	FT administrator only
ftdell	Delete log record	FT or FTAC administrator only
ftdelp	Delete FT profile	FT user also [1)]
ftexpe	Export FT profiles and admission sets into file	FTAC administrator only
ftimpe	Import FT profiles and admission sets from a file into the local system	FTAC administrator only
ftlang	Set language interface (must be called with `/opt/openFT/bin/ftbin/ftlang`)	FT administrator only
ftmoda	Modify admission set	FT user also [1)]
ftmodi	Modify an instance	FT administrator only
ftmodo	Modify operating parameters	FT administrator only
ftmodp	Modify FT profile	FT user also [1) 3)]
ftmodptn	Modify partner attributes	FT administrator only
ftmodr	Change the order of the requests in the request queue	FT user also [1)]
ftremptn	Remove partner from the partner list	FT administrator only
ftsetjava		FT administrator only
ftshwa	Display admission sets	FT user also [1)]
ftshwd	Display diagnostic records	FT or FTAC administrator only

Command	Function	Note
ftshwe	Display FT profiles and admission sets from file	FTAC administrator only
ftshwl	Display log records	FT user also [1]
ftshwo	Display operating parameters	FT user also
ftshwp	Display FT profiles	FT user also [1]
ftshwptn	Display partner	FT user also
ftshwr	Display properties and statuses of requests	FT user also [1] [2]
ftstart	Start asynchronous *open*FT server	FT administrator only
ftstop	Stop asynchronous *open*FT server	FT administrator only
ftupdi	Update the instance directory	FT administrator only
ftupdk	Update the public keys	FT administrator only
install.ftam	Install *open*FT-FTAM	FT administrator only
install.ftp	Install *open*FT-FTP	FT administrator only

1) This command is also available to users with restricted functional scope.

2) This command is described only in the User Guide.

3) This command is described in detail in the User Guide for *open*FT. This manual describes only the switches and values that offer you additional options as an administrator.

As the **administrator**, you may execute the commands listed below with the additional options to perform the corresponding action **system-wide**. This means that:

You can use *ftcanr* to delete any desired file transfer requests.

You can use *ftcrep* to create FT profiles for any login names

You can use *ftdelp* to delete any FT profiles.

You can use *ftmoda* to modify any of the admission sets.

You can use *ftmodp* to modify any of the FT profiles.

You can use *ftmodr* to change the order of all requests in the request queue independent of the login name.

You can use *ftshwa* to display any of the admission sets.

You can use *ftshwl* to display any of the log records.

You can use *ftshwp* to display any of the FT profiles.

You can use *ftshwr* to obtain information about all the requests for all user IDs.

5.2 Notational conventions

The command syntax essentially corresponds to the output that you get when you specify the command with *-h* option. The following conventions have been used for syntax diagrams:

< > angle brackets are used for parameters which you may replace with current values. You must not specify the angle brackets < > and the permissible value ranges.

[ ] enclose optional entries. The effect on the function of the command is described for the individual parameters.

␣ stands for at least one blank that must be inserted between the various entries.

| stands for alternatives. You may specify only one of the values indicated.

Lengths and characters sets

The values which you use for parameters in the commands must observe certain restrictions on length and on the characters available:

file name

you can specify an absolute or relative file name. The file name specified in the local and remote systems may have a maximum length of 512 characters based on the length of the absolute path name. Please note that although long file names can be specified at the *open*FT interfaces, not all platforms support this maximum length. For example UNIX systems permit up to 512 characters whereas Windows systems only permit 256 characters.

If the file name contains blanks, they must be set in quotation marks (e.g. "file name"). If the remote system requires quotation marks around the file name, these must not be canceled (e.g. ‘file name’) as on the shell level.

date

numeric; exactly 8 characters in the form `yyyymmdd` with:
`yyyy` for year, `mm` for month and `dd` for day

Note that for all date entries, you may only specify values up to and including 20380119 (January 19, 2038)

user ID

User ID for accessing the required system, maximum 64 characters + 3 characters for hexadecimal format (X' '). The maximum length is system-dependent:
In UNIX systems, a maximum of 32 characters with first 8 characters being unique; in Windows systems, a maximum of 36 characters.

command

up to 1000 characters; for follow-up processing commands, the commands for success and failure must not be longer than 1000 characters in total.

partner address

The address of the partner system is output in the following form:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

protocol

protocol stack via which the partner is addressed. Possible values:

openft (*open*FT protocol), default value

ftam (FTAM protocol)

ftp (ftp protocol)

host internet host name, IP address or GLOBAL NAME from the TNS, mandatory parameter. Format of the IP addresses (example):
`%ip111.222.123.234` (ipv4) or
`%ip6[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]` (ipv6) or
`%ip6[FE80::20C:29ff:fe22:b670%5]` (ipv6 with Scope-ID)

The brackets [..] are mandatory with ipv6.

port port number in the case of a TCP/IP connection, optional.

tsel transport selector (only *open*FT and FTAM protocol), optional.

ssel session selector for FTAM connection or session routing information in the case of *open*FT protocol with *open*FTIF connection, optional.

psel presentation selector for FTAM connection, optional.

profile name

alphanumeric (a..z, A..Z, 0..9), up to 8 characters

transfer admission

the transfer admission usually consists of printing characters and may not start with a hyphen, minimum 8 characters, maximum 67 characters (in UNIX systems, maximum 32 characters). If a transfer admission consists of non-printing characters then it must be specified in hexadecimal format in the form x\'...\' or X\'...\'.

Special characters in the entries for *file name, file name-prefix*, *transfer admission*, *user ID*, *account*, *password*, *follow-up processing* (see notes on the commands) must be escaped using a backslash (\). Here, you must differentiate between special characters for file transfer and special characters on a UNIX based operating system, and escape the special characters accordingly.

Example

The account number 1111111,00000000,88888888 is specified in the transfer admission. The comma is a special character that enables file transfer to distinguish the elements of the triple, and must therefore be escaped with a backslash (\). This reverse slash is also a special character for the shell, and must therefore also be escaped. The entry then appears as follows:

“1111111\\,00000000\\,88888888”

The **sequence** of entries in the command is arbitrary, except for the entries for
– the source and destination of a request (e.g. local and remote file name, partner name,...)
– the authorization to access the remote system, i.e., the transfer admission or the system login.

When there is a large number of parameters, *open*FT commands can be very long. If you want to use the keyboard to enter commands that are longer than 256 characters, you will need to work with continuation lines. You can obtain these by entering the sequence \ ↵.

Note that the entries for follow-up processing must be enclosed in single or double quotes (' or ").

If the entry for follow-up processing also contains single quotes (’), it is recommended to enclose the entire entry in double quotes ("). The single quotes in the follow-up processing command (e.g. single quotes in a BS2000 password) can then be written as expected in the partner system (such as BS2000).

Some FT commands have a very extensive syntax. You can also have the syntax of any given command displayed on the screen using the *-h* option.

5.3 Output in CSV format

For some Show commands, *open*FT for UNIX systems offers output in CSV format. CSV (**C**omma **S**eparated **V**alues) is a popular format in the PC environment in which tabular data is defined by lines. Output in CSV format is offered for the following commands:

– ftshw
– ftshwa
– ftshwe
– ftshwl
– ftshwo
– ftshwp
– ftshwptn
– ftshwr

Output in CSV format is also possible for the *open*FT-Script commands ftshwact and ftshws, see User Guide.

Many programs such as spreadsheets, databases, etc., can import data in CSV format. This means that you can use the processing and presentation features of such programs on the data output by the above commands.

The output fields are described in the appendix starting on page 179 and in the User Guide.

Every record is output as a line, and each record contains information on an object. The first line is always the header and contains the field names of the respective columns. **Only the field names are guaranteed, not the order of fields in a record**. In other words, the order of columns is determined by the order of the field names in the header line. Fields within an output line are separated by semicolons (;).

The following data types are differentiated in the output:

Number

String
: Since the ";" (semicolon) character has a special meaning in the CSV output as a field separator, any text containing a ";" is enclosed within double quotes.
Keywords are never enclosed within double quotes and **always** begin with the character "*" (asterisk).

Date

Date and time are always output in the format `yyyy-mm-dd hh:mm:ss`; a date alone is output in the format `yyyy-mm-dd`.

One example of a possible evaluation procedure is supplied as a reference template in the Microsoft Excel format in the file */opt/openFT/samples/ftaccnt.xlt*. The template evaluates a CSV log file by means of an automatically running macro. The result shows the number of inbound and outbound requests and the Kilobytes transferred in each case for all users.

5.4 ftaddptn - Enter a partner in the partner list

You use the *ftaddptn* command to enter a partner system in the local system's partner list.

Format

ftaddptn -h |
[<partner name 1..8>]
-pa=<partner address 1..200>
[-id=<identification 1..64> | -id=]
[-ri=<routing info 1..8> | -ri=@i | -ri=]
[-ptc=i | -ptc=a | -ptc=]
[-sl=1..100 | -sl=p | -sl=]
[-st=a | -st=d | -st=ad]
[-am=y | -am=n]
[-tr=n | -tr=f | -tr=]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner name
This is the name to be used to enter the partner system in the partner list. The name may consist of 1 to 8 alphanumerical characters. The first character must be a letter and no distinction is made between uppercase and lowercase. The name can be chosen freely and need only be unique within *open*FT.

-pa=partner address
You use *-pa* to enter the address of the partner system in the following form:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

protocol
Protocol stack via which the partner is addressed.
Possible values:

openft
openFT partner, i.e. communication via the *open*FT protocol

ftam FTAM partner, i.e. communication via the FTAM protocol

ftp ftp partner, i.e. communication via the ftp protocol

Default value: **openft**

Exception: if a global name from the TNS is used for *host* and a presentation selector is assigned to this global name then *ftam* is the default value. Please note that TNS use must be activated for this (*ftmodo -tns=y*).

host Internet host name, IP address or GLOBAL NAME from the TNS, mandatory parameter. Format of IP addresses (example):
`%ip111.222.123.234` (ipv4) or
`%ip6[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]` (ipv6) or
`%ip6[FE80::20C:29ff:fe22:b670%5]` (ipv6 with Scope-ID)

The brackets [..] are mandatory with ipv6.

port Port number in the case of a TCP/IP connection, optional.

tsel Transport selector (only *open*FT and FTAM protocol), optional.

ssel Session selector for FTAM connection or session routing information in the case of *open*FT protocol with *open*FTIF connection, optional.

psel Presentation selector for FTAM connection, optional.

For details concerning address specifications, see *open* FT User Guide.

-id=identification | **-id=**

Identification unique in the network of the *open*FT instance in the partner system. In the case of FTAM partners, it is possible to specify an Application Entity Title in the form n1.n2.n3.n4..mmm as the identification. n1, n2 etc. are positive integer values which describe the "Application Process Title". n1 can only have the values 0, 1 or 2, n2 is restricted to values between 0 and 39 if n1 does not have the value 2. The optional Application Entity Qualifier mmm must be separated from the values of the Application Process Title by two periods. For details, see the *open*FT User Guide.

Identification not specified

The specification of *-id=* means that the *host* (host name) is used for identification.

Standard value: *host* (host name) for the *open*FT protocol, otherwise blank.

-ri=routing info | **-ri**=@i | **-ri=**

If the partner system can only be accessed via an intermediate instance (e.g. *open*FTIF gateway) then you specify the address information to be used for routing by the intermediate instance in *routing info*.

@i for *routing info*

The instance identification specified in *-id=* is used as the routing information.

neither *@i* nor *routing info* specified (default value)

The specification of *-ri=* (without parameters) means that the partner system can be accessed directly, i.e. without an intermediate instance.

-ptc=i | **-ptc**=a

You can use *-ptc* to modify the operating parameter setting for sender verification on a partner-specific basis. These settings only affect partners which are connected via the *open*FT protocol and do not operate with authentication (e.g. partners with *open*FT V8.0 or earlier).

i (identification)

Deactivates checking of the transport address. Only the partner's identification is checked. The partner's transport address is also not checked even if extended sender verification is globally active (see the *ftmodo* command on page 106).

a (address)

Activates checking of the transport address. The partner's transport address is checked even if checking of the transport address is globally deactivated (see *ftmodo* command on page 106).

If the transport address under which the partner logs on is not the same as the entry in the partner list then the request is rejected.

neither *i* nor *a* specified (default value)

-ptc= (without parameters) means that the operating system parameters apply to sender verification.

-sl=1..100 | **-sl**=p | **-sl=**

You use this option to assign a security level to the partner system.

1..100

Assigns a fixed security level to the partner. 1 is the lowest and 100 the highest security level.

p Assigns a security level to the partner on the basis of the partner's attributes, i.e.:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system.
- Security level 100 otherwise, i.e. if the partner has only been identified by its address.

Security level not specified (default value)
-sl= (without parameters) means that the operating parameter setting for the security level applies (see command *ftmodo* on page 106).

-st=a | **-st**=d | **-st**=ad
This option allows you to control how locally submitted asynchronous file transfer requests to the specified partner system are processed.

a (active, default value)
Locally submitted asynchronous file transfer requests to this partner system are processed if the asynchronous *open*FT server is started.

d (deactive)
Locally submitted asynchronous file transfer requests to this partner system are initially not processed but are stored in the request queue.

ad (automatic deactivation)
Unsuccessful attempts to establish a connection to this partner system result in its deactivation. The maximum number of unsuccessful attempts is 5. If you want to perform file transfer again with this system, you must explicitly activate it with *ftmodptn -st=a*.

-am=n | **-am**=y
You can use this option to force partner authentication.

n (default value)
Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

y Authentication is forced, i.e. requests are only processed if the local system is successfully able to authenticate the partner, see page 27.

-tr=n | **-tr**=f | **-tr**=

You can use this option to modify the operating parameter settings for the partner selection for the *open*FT monitoring function on a partner-specific basis.

n (on)

The monitoring function is active for this partner. However, a trace is only written if the *open*FT monitoring function has been activated via the operating parameters, see page 106ff, *ftmodo*, option *-tr*.

f (off)

The monitoring function is deactivated for this partner.

neither *n* nor *f* specified (default value)

-tr= (without parameters) means that the global setting for partner selection in the *open*FT monitoring function applies (see *ftmodo* command on page 106).

5.5 ftalarm - Report failed requests

The *ftalarm* command is used to trigger an alarm if, within two minutes, more FT requests than the number specified by the user fail. The failed FT requests are identified using the return code not equal to 0 for the FTAC log records. *ftalarm* uses the *cron* function.

A separate *ftalarm* call is required for each instance.
Proceed as follows: activate the instance with *ftseti*, and call *ftalarm*.

Format

ftalarm [-h |
-s <number of errors 1..99999999> |
-t]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-s <number of errors>
starts the *ftalarm* function. When the specified *number of errors* in FTAC log records is exceeded within two minutes, the following message is output on the console and to the file */var/openFT/instance/log/conslog (*where *instance* means the name of the corresponding instance):

```
openFTalarm: number or more access control error loggings
within 2 minutes
```

The partial string *openFTalarm:* within this message is also guaranteed for future versions of *open*FT and can be interpreted for automatic processing by system management tools.

The messages are output by the *cron* function at regular intervals and can therefore be delayed by up to one minute when the *ftalarm* function is activated.

-t terminates the *ftalarm* function.

5.6 ftcrei - Create or activate an instance

The *ftcrei* command allows you to create a new instance or re-activate a deactivated instance.

When an instance is created, the instance file tree is linked to the */var/openFT* directory with the resources of an instance.

If the specified instance file tree does not yet exist, it is created.

When the instance file tree is created, the operating parameters, the profile files and the startup and shutdown files are initialized in the same way as for a new installation.

If the instance file tree already exists, *ftcrei* checks the version. If the instance file tree was created using an older version of *open*FT, it must first be updated using the *ftupdi* command before it can be reactivated.

Important notes for when using multiple instances

– Use of several *open*FT instances is only possible using the TCP/IP transport system. If you would like to use several instances and are working with TNS (*ftmodo -tns=y*), you must delete all openFT-specific TNS entries that are not TCP/IP compliant (i.e. all except for LANINET and RFC1006).

– You must explicitly assign an individual address to all instances using *-addr*.

– If the instance is to be authenticated in partner systems, it must have a unique instance ID assigned to it (using *fta -id*=). In addition, a public key for the instance must be made available to the partner systems.

Format

ftcrei -h |
<instance 1..8> [<directory 1...128>] [-addr=<host name>]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance
Name of the instance to be created.
Instance names have a maximum length of 8 characters and must con-

sist of alphanumeric characters. The first character must not be a number. The instance name must not be confused with the instance ID (see *ftmodo -id=*).

directory
: Directory in which the instance file tree is to be located.
By default, it is created in:

`/var/openFT/`*instance*

-addr=host name

: Internet host name by which the instance is addressed. If your system has a DNS name, you should specify the full DNS name. *open*FT then uses the first 8 characters of the first part of the name (the host name qualifier) as the processor name (*ftmodo -p=*) and the entire name as the instance ID (*ftmodo -id=*).

Messages of the ftcrei command

If *ftcrei* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

Examples:

1. The instance *inst1* is to be newly created in the directory */cluster/inst1*. The DNS name is *hugo.abc.net*. The directory */cluster/inst1* is not allowed to exist.

   ```
   ftcrei inst1 /cluster1/inst1 –addr=hugo.abc.net
   ```

   Where the operational parameter *ftmodo -p=* is *hugo* and *ftmodo -id=* is *hugo.abc.net*.

2. The existing instance *inst2* from the directory */cluster/inst2* is to be re-activated. No host name may be specified.

   ```
   ftcrei inst2 /cluster/inst2
   ```

5.7 ftcrek - Create key pair set

You use this command to create a key pair set for the authentication of your *open*FT instance in partner systems (RSA procedure). For more information on administering keys, see the section “Authentication” on page 27.

Format

ftcrek [-h]

Description

-h Displays the command syntax.

5.8 ftcrep - Create an FT profile

ftcrep stands for "create profile". This command can be used by any user to set up FT profiles for his or her login name.

The FTAC administrator can also set up FT profiles for other login names, either with or without defining a transfer admission.

Format

```
ftcrep  -h |
          <profile name 1..8>
          <transfer admission 8..32> | @n
        [ -ua=[ <user ID 1..32> ] [, [ <password 1..20> | @n ] ] ]
        [ -v=y | -v=n ] [ -d=yyyymmdd ]
        [ -u=pr | -u=pu ]
        [ -priv=y | -priv=n ]
        [ -iml=y | -iml=n ]
        [ -iis=y | -iis=n ] [ -iir=y | -iir=n ]
        [ -iip=y | -iip=n ] [ -iif=y | -iif=n ]
        [ -ff=t | -ff=m | -ff=r | -ff=p | -ff=tmpr | -ff=prmt ]
        [ -dir=f | -dir=t | -dir=ft ]
        [ -pn=<partner 1..200>,...,<partner(50) 1..200> | -pn= ]
        [ -fn=<file name 1..512> | -fn= ]
        [ -fnp=<file name prefix 1..511> ]
        [ -ls= | -ls=@n | -ls=<command1 1..1000> ]
        [ -lsp=<command2 1..999> ] [ -lss=<command3 1..999> ]
        [ -lf= | -lf=@n | -lf=<command4 1..1000> | ]
        [ -lfp=<command5 1..999> ] [ -lfs=<command6 1..999> ]
        [ -wm=o | -wm=n | -wm=e | -wm=one ]
        [ -c=y | -c=n]
        [ -txt=<text 1..100> ]
```

Description

In the following, only those switches and values are described, which are very important for the administrator or which offer the administrator additional options. For the other options, see the User Guide.

profile name
: is the name you wish to assign to the FT profile. This name can be used to address the FT profile, for example when it is to be modified or deleted. Be sure not to confuse the profile name with the transfer admission (see

below). The profile name must be unique among all the FT profiles under your login name, or FTAC will reject the *ftcrep* command and issue the message FT profile already exists. To have the profile names you have already assigned displayed, you can issue the *ftshwp* command (without options).

transfer admission | @n
: replaces the login authorization for your UNIX system otherwise required in FT requests. When this transfer admission is specified in an FT request, FTAC applies the access rights defined in this FT profile.

  transfer admission
  : The transfer admission must be unique within your UNIX system so that there are no conflicts with transfer admissions defined by other FTAC users with other access rights. If the transfer admission you select has already been assigned, FTAC rejects the *ftcrep* command and issues the message:
    `Transfer admission already exists.`

    You can also define a binary admission with any characters, including non-printing characters. To do this, you must specify the transfer admission in hexadecimal format in the following form: x\'...\' or X\'...\', z.B. x\'f1f2f3f4f5f6f6f8\'.

    As the FTAC administrator, you can assign a transfer admission for yourself under your own login name or for any other user. In this case, however, you must specify the entire login authorization (i.e. the login name and password for that login name) if you do not have root privileges (UID=0).

  @n for *transfer admission*
  : As the FTAC administrator, by specifying @n, you can create FT profiles for other login names without having to define transfer admissions. The owner of the login name for which the FT profile was created can then enable this profile using the *ftmodp* command. In order to do this, the owner must specify a transfer admission with *ftmodp*.

transfer admission not specified
: FTAC will then prompt you to enter the transfer admission. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program expects you to enter the transfer admission a second time as an entry check.

-ua=[user ID][**,**[password | @n]]

FTAC administrators use *-ua* to specify the user IDs for which they want to set up FT profiles.

user ID
: The user can specify only his own user ID.
As the FTAC administrator, you can specify any user ID.

,password
: Specifies the password of the login name. A binary password must be specified in hexadecimal format in the form x\'...\' or X\'...\'. The FT profile for the login name is only valid while the password is valid for the login name. If the password is changed, the profile can no longer be used.

  If you want to assign an FT profile for another user and also assign a transfer admission for that profile, you must specify the login name as well as the password for that login name if you do not have root privileges (UID=0).

@n for *password*
: This entry may only be specified by the FTAC administrator. With *@n*, you cannot assign any transfer admission for the FT profile if you do not have root privileges (UID=0).

comma only (,) no *password* specified
: causes FTAC to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the transfer admission must be entered a second time.

user ID only (without comma and no *password*) specified
: the profile is valid for all the passwords for *user ID*.

-ua=␣ specified or *-ua* not specified
: the FT profile is created for the individual login name.

-priv=n | -priv=y
: is used by the FTAC administrator to grant privileged status to FT profiles.
  n (default value)
  : The FT profile is not privileged (initially).

  y The FT profile is privileged.

-iml=y | -iml=n
-iis=y | -iis=n
-iir=y | -iir=n
-iip=y | -iip=n
-iif=y | -iif=n

These options are used to specify whether the FT profile is to be restricted by the values in the admission set (MAX. USER LEVELS). If the FT profile is also privileged by you as the FTAC administrator, the entries you have made (the MAX. ADM LEVELS) can also be ignored. This FT profile would then allow *inbound* basic functions which are disabled in the admission set to be used. Possible values are:

y allows the values in the admission set to be ignored.

n (default value)
restricts the functionality of the profile to the values in the admission set.

The following table shows which subcomponents of the file management can be used under which conditions.

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled
Modify file attributes	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Rename files	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Delete files	Inbound Receive(IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management(IBF) enabled
Create, rename and delete directories	Inbound File Management(IBF) enabled **and** direction= from partner in profile

5.9 ftdeli - Delete or deactivate an instance

The *ftdeli* command allows you to delete an instance. Deleting an instance removes only the symbolic link in the local */var/openFT* directory. The instance file tree is not changed. The standard instance *std* and the currently set instance can not be deleted.

Format

ftdeli -h |
<instance 1..8>

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance
Name of the instance to be deleted.
Instance names have a maximum length of 8 characters and must consist of alphanumeric characters. The first character must not be a number.

Messages of the ftdeli command

If *ftdeli* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

Examples

1. The instance *inst1* from the directory */CLUSTER/inst1* is to be deactivated on computer *CLUSTER1*, since it has been switched over to *CLUSTER2*. The directory */CLUSTER/inst1* is retained.

```
ftdeli inst1
```

2. Instance *inst2* with the directory */CLUSTER/inst2* is to be deleted along with the instance file tree.

```
ftdeli inst2
rm -r /CLUSTER/inst2
```

3. Using *ftseti*, it was changed to instance *inst3*. There, an attempt is being made to deactivate the instance *inst3*.

```
ftdeli inst3
ftdeli: openFT Instance 'inst3' can not be removed.
```

5.10 ftdelk - Delete key pair set

You use this command to delete the key pair sets for a reference. Your system can then no longer be authenticated by partner systems which still use the associated public key. For more information on administering keys, see section “Authentication” on page 27.

A key pair set should always be present in your *open*FT instance as otherwise all requests are run unencrypted, i.e. neither the request data nor the file contents are encrypted.

Format

ftdelk [-h] <key reference 1..9999999>

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

key reference
Used to select the key pair set that is to be deleted. You can find the reference in the name of the public key file, see section “Creating and administering RSA key pairs” on page 29.

5.11 ftdell - Delete log record

With *ftdell*, you can delete FT log records for all login names This function is not permitted for the ordinary user.

Store the log records by redirecting the output of *ftshwl* to a file or to the printer (see section "ftshwl - Display log records" in the user manual).

Deleting log records changes the size of the file since the storage space is freed immediately after deletion.

The time by which the log records are to be deleted can be entered either as a fixed time with date and time or as a relative time; for example: all records before 10 days ago.

By default, *open*FT deletes all log records which are older than 30 days every time the system is started up.

Format

ftdell -h |
[-rg=[[[yyyy]mm]dd]hhmm | -rg=#1..99999999 | -rg=0..999]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-rg=[[[yyyy]mm]dd]hhmm
You use *-rg* to specify the end of a logging interval.
When selecting the time, this is interpreted as as follows:
- a 4-digit specification is interpreted as the time expressed in hours and minutes,
- a 6-digit specification as the day (date) and time in hours and minutes,
- an 8-digit specification as the month, day, and time in hours and minutes,
- a 12-digit specification as the year, month, day, and time in hours and minutes.

The largest possible value that can be specified as the date is 20380119 (January 19, 2038).
*open*FT then deletes all log records which are older that the specified time.
The optional data ([…]) is automatically replaced by current values.

-rg=#1..99999999
: Here you use *-rg* to specify the end log ID. It is identified by a leading # character, followed by the 1-8-digit ID.
*open*FT then deletes all log records which belong to this log ID or which belong to a smaller log ID.

-rg=0..999
: Here you use *-rg* to specify a time interval (relative to the current date and time) as a multiple of 24 hours, i.e. number of days. The specification can be 1-3 digits long.
*open*FT then deletes all log records which are older than the specified time. This means you are looking back in time. If you specify *-rg=2*, for example, all log records which are older than two days (48 hours) are deleted.

-rg not specified
: The range is not a selection criterion, i.e. all log records are to be deleted by 00:00 hours of the current date.

Example

1. As the FT or FTAC administrator, you wish to delete all FT log records written up to 00:00 hours of the current date.

   ```
   ftdell
   ```

2. As the FT or FTAC administrator, you wish to delete all FT log records written up to the current time:

   ```
   ftdell -rg=0
   ```

3. As the FT or FTAC administrator, you wish to delete all log records written before the last 7-day period (7 times 24 hours before the current time:

   ```
   ftdell -rg=7
   ```

4. As the FT or FTAC administrator, you wish to delete all log records from the beginning to the record with the log ID 1450:

   ```
   ftdell -rg=#1450
   ```

5.12 ftdelp - Delete FT profiles

ftdelp stands for "delete profile". When checking your set of profiles (with *ftshwp*), it is a good idea to weed out, from time to time, those which are no longer needed and are merely taking up storage space.

ftdelp allows the FTAC administrator to delete FT profiles belonging to other login names as well. Of course, the administrator should first inform the owner of the profiles before deleting them.

Format

ftdelp -h |
<profile name 1..8> | @a
[-s=[<transfer admission 8..32>| @a |@n]
[,<user ID 1..32> | @a]]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name | @a
is the name of the FT profile you wish to delete.

@a for *profile name*
profile name is not used as a criterion for selecting the FT profile to be deleted. If you do not identify the profile more closely with *-s* (see below) you will delete all of your FT profiles.

-s=[transfer admission | @a | @n][,user ID | @a]
-s is used to specify criteria for selecting the FT profiles to be deleted.

transfer admission
is the transfer admission of the FT profile to be deleted. A binary transfer admission must be specified in the form x\'...\' or X\'...\'.

@a for *transfer admission*
deletes either the FT profile specified by *profile name* (see above) or all of your FT profiles.

As the FTAC administrator, you must specify *@a* if you want to delete FT profiles belonging to other login names, since you actually should not know the transfer admission.

@n for *transfer admission*

As the FTAC administrator, you can specify *@n* if you only want to delete FT profiles of other login names, which do not have any defined transfer admissions.

transfer admission not specified

causes to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time. If you just press ↵, this has the same effect as specifying *@a*.

,user ID

As the FTAC administrator, you can specify any login name.

@a for *user ID*

If you specify *@a* as the FTAC administrator, FT profiles belonging to all login names are deleted.

user ID not specified

deletes only profiles belonging to the user's own login name, regardless of who issues the command.

-s not specified

if *@a* is specified for *profile name*, all the FT profiles belonging to the login name under which the *ftdelp* command is issued are deleted. Otherwise, the FT profile with the specified name is deleted.

5.13 ftexpe - Export FT profiles and admission sets

ftexpe stands for "export environment", i.e. exporting the FTAC environment, or exporting FT profiles and admission sets.

Using *ftexpe* the FTAC administrator can write FT profiles and admission sets of any login names to files, thereby saving them.

However, the standard admission set is not saved and the variable values in an admission set (values marked with an asterisk (*)) that refer to the standard admission set, are saved as variables. This means that there is no fixed value for the relevant basic function in the backup. If an admission set is imported, the relevant basic function receives the value of the standard admission set that is currently valid.

FT profiles and admission sets saved in this way can be re-imported using the *ftimpe* command.

Format

ftexpe -h |
<file name 1..512>
[-u=<user ID 1..32>[,...,<user ID(100) 1..32>]]
[-pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n]
[-as=y |-as= n]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name
With *file name*, you specify the name of the file in which the FT profiles and records are to be written. You may access this file only using the *ftimpe* and *ftshwe* commands. *path/file name* must not be longer than 512 characters, and no backup files with the same name must exist in the current directory.

-u=user ID1[,user ID2][,user ID3]...
-u specifies the login names whose FT profiles and admission sets are to be saved to a file. Up to 100 login names can be specified simultaneously.

-u not specified
: all FT profiles and admission sets on the system are saved to the specified file.

-pr=profilname1[,profilname2][,profilname3]... | @n
: specifies the FT profiles to be saved to the specified file (up to 100).

@n for *profilname*
: no FT profiles are saved.

-pr not specified
: all FT profiles belonging to the login names specified in the -u parameter, are saved.

-as=y |-as= n
: specifies whether or not the admission sets should be saved to the specified file. Possible values are:

y (default value)
: all admission sets belonging to the login names specified in the *-u* parameter, are saved.

n
: no admission sets are saved.

Example

The admission set and the FT profiles belonging to the login name *donald* are to be saved. *ftacsave* is specified for the backup file.

```
ftexpe␣ftacsave␣-u=donald
```

5.14 ftimpe - Import profiles and admission sets

ftimpe stands for "import environment", i.e. importing the FTAC environment or importing FT profiles and admission sets. Using *ftimpe*, the FTAC administrator can import the FT profiles and admission sets of any login names from a file that was created using the *ftexpe* command.

Only those FT profiles whose profile names have not been specified for other FT profiles under the specified login name are imported.

An FT profile whose transfer admission has already been defined for another FT profile in the system will be imported, but has an undefined transfer admission. It must therefore be assigned a new transfer admission using the *ftmodp* command before it is used. If the existing FT profile in the system is designated as private, it is immediately disabled. It must be assigned a new transfer admission using the *ftmodp* command, before it is used.

The imported FT profiles are automatically locked and must be unlocked before use with the command *ftmodp* and the parameter *-v=y* if the FTAC administrator does not have root privileges (UID=0). Privileged FT profiles lose their privileged status when imported. The FTAC administrator can control this behavior with the *-sec* option provided that he has root privileges.

The standard admission set is not saved when it is exported. Therefore, the standard admission set on the computer at the time of importing remains valid. Variable values in the imported admission sets, that refer to the standard admission set (and are therefore marked with an asterisk (*), are assigned the value of the standard admission set that is currently valid.

Format

```
ftimpe -h  |
          <file name 1..512>
        [ -u=<user ID 1..32>[,...,<user ID(100) 1..32>] ]
        [ -pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n ]
        [ -as=y |-as=n ]
        [ -sec=s | -sec=h ]
```

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

file name specifies the file from which the FT profiles and admission sets are to be imported.

-u=user ID1[,user ID2][,user ID3]...

specifies the login names whose FT profiles and admission sets are to be imported. You can specify up to 100 login names simultaneously.

-u not specified

all FT profiles and admission sets are imported.

-pr=profile name1[,profile name2][,profile name3]...| -pr=@n

specifies the FT profiles to be imported (up to 100).

@n for *profile name*

no FT profiles are imported.

-pr not specified

all FT profiles belonging to the login names specified in the *-u* parameter are imported. However, the profile is not imported if another FT profile of the same name already exists under this login name.

-as=y | **-as**= n

specifies whether or not admission sets are to be imported. Possible values are:

y (default value)

all admission sets belonging to the login names specified in the *-u* parameter are imported.

n no admission sets are imported.

-sec=s | **-sec**=h

-sec specifies the security level when importing FT profiles. It only makes sense to use the *-sec* option if you, the FTAC administrator, have root privileges (UID=0).

s (standard) If you have root privileges, the attributes of the FT profile are not changed when it is imported.

If you do not have root privileges, the effect is the same as *-sec=h*, i.e. the profiles are locked.

-sec=s is the default value.

h (high) The FT profiles are locked (LOCKED (by import)) and are assigned the attributes *private* and *not privileged*.

Example

The admission set and FT profiles of the login name *donald* were saved to the file *ftacsave* with *ftexpe*. They are to be imported to another system under the same login name.

```
ftimpe␣ftacsave␣-u=donald
```

As the FTAC administrator you may receive the following messages, for example:

```
OWNER    NAME
donald   secret1     FT profile already exists.
         secret2
```

These messages indicate that *donald* has already created the FT profiles *secret1* and *secret2* on the new system, and these profiles were therefore not imported.

Note

If, after import, you wish to delete an admission set for a login name that does not exist on your computer, enter the command *ftmoda ␣login-name ␣-ml=s*. This situation can occur when you use *ftexpe* to incorporate into your system a file that has been created on a different host.

5.15 ftlang - Change default language setting

The default language for *open*FT is determined by evaluating the LANG environment variable during installation.

You can switch languages later on using the shell procedure */opt/openFT/bin/ftbin/ftlang*.
For more details see section “Switching the language interface” on page 23.

Format

ftlang [-h |
-i |
de |
en]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i you can use this switch to query the currently set language variant.

de *open*FT is switched to German as the default.

en *open*FT is switched to English as the default.

In both cases, the necessary messages files, the *fthelp* procedure, the manpages and the help texts of the graphical user interface are activated.

Example

The default language setting is switched from German to English:

```
/opt/openFT/bin/ftbin/ftlang␣en
```

5.16 ftmoda - Modify admission sets

ftmoda stands for "modify admission set". As the FTAC administrator, you can use this command to define settings for the standard admission set and for any admission set of any user in the system. The settings made by the administrator for other users are the MAX. ADM LEVELS.

You can assign a security level of between 0 and 100 for each basic function. These values have the following meanings:

0	The basic function is locked, i.e. it is not released for any partner system.
1 to 99	The basic function is only released for partner systems with the same or a lower security level. You can use the *ftshwptn* command to display the security level of a partner system.
100	The basic function is available for all partner functions.

For basic functions, consult the table on page 104.

The FTAC administrator can also use *ftmoda* to make another login name the FTAC administrator.

The meaning of the numbers in the admission set has been changed in *open*FT V10.0. Now, all integers between 0 and 100 are analyzed and are compared with the partner system security levels to determine whether they are smaller than or equal to these values.

Format

```
ftmoda -h |
        [ <user ID 1..32> | @s ]
        [ -priv=y ]
        [ -ml=s | -ml=0..100 ]
        [ -os=s | -os=0..100 ]
        [ -or=s | -or=0..100 ]
        [ -is=s | -is=0..100 ]
        [ -ir=s | -ir=0..100 ]
        [ -ip=s | -ip=0..100 ]
        [ -if=s | -if=0..100 ]
```

Description

-h Displays the command syntax on the screen. Entries after the are ignored.

user ID | @s
: As the FTAC administrator, you can specify any login name desired.

  @s for *user ID*
  : By entering the value @s, the FTAC administrator can modify the standard admission set.

user ID not specified
: modifies the admission set of the login name under which *ftmoda* is entered.

-priv=y
: As the FTAC administrator, you can assign administrator privileges to the specified *user ID*.

-priv not specified
: does not change the FTAC administrator.

-ml=s | -ml=0..100
: sets the same value for all six basic functions.

  Possible values are:

  s sets each of the basic functions to the value defined in the standard admission set.

  0 disables all of the basic functions.

  1 .. 99
  : All basic functions are released only for partner systems with an equal or lower security level.

  100 All basic functions are released for all partner systems. For outbound file management functions, no check is made.

-ml not specified
: leaves the settings in the admission set unchanged if none of the following entries are made.

-os=s | -os=0..100
sets the value for the basic function *outbound send*, which determines whether or not requests initiated in your UNIX system can send data to a remote system.

s sets the value defined in the standard admission set.

0 disables the basic function *outbound send*.

1 .. 99
The basic function *outbound send* is released only for partner systems with an equal or lower security level.

100 enables the basic function *outbound send* for all partner systems.

-os not specified
leaves the setting for *outbound send* unchanged.

-or=s | -or=0..100
sets the value for the basic function *outbound receive*, which determines whether or not requests initiated in your UNIX system can fetch data from a remote system.

s sets the value defined in the standard admission set.

0 disables the basic function *outbound receive*.

1 .. 99
The basic function *outbound receive* is released only for partner systems with an equal or lower security level.

100 enables the basic function *outbound receive* for all partner systems.

-or not specified
the value for *outbound receive* is not changed.

-is=s | -is=0..100
sets the value for the basic function *inbound send*, which determines whether or not a remote partner system can fetch data from your UNIX system.

s sets the value defined in the standard admission set.

0 disables the basic function *inbound send*.

The subcomponent of the basic function *inbound file management* for "displaying file attributes" is also disabled.

Any requests from remote FTAM partner systems to create a new file are also rejected.

1 .. 99
The basic function *inbound send* is released only for partner systems with an equal or lower security level.

100 enables the basic function *inbound send* for all partner systems.

-is not specified
leaves the setting for *inbound send* unchanged.

-ir=s | -ir=0..100
sets the value for the basic function *inbound receive*, which determines whether or not a remote partner system can send data to your UNIX system.

s sets the value defined in the standard admission set.

0 disables the basic function *inbound receive*.

The subcomponents of the basic function *inbound file management*, deletion and renaming of files, as well as modification of file attributes, are also locked.

1 .. 99
The basic function *inbound receive* is released only for partner systems with an equal or lower security level.

100 enables the basic function *inbound receive* for all partner systems.

-ir not specified
leaves the setting for *inbound receive* unchanged.

-ip=s | -ip=0..100
sets the value for the basic function *inbound follow-up processing + preprocessing + postprocessing*, which determines whether or not a remote system may request follow-up, pre- or postprocessing on your UNIX system.

s sets the value defined in the standard admission set.

0 disables the basic function *inbound follow-up processing + preprocessing + postprocessing*.

1 .. 99
The basic function *inbound follow-up processing + preprocessing + postprocessing* is released only for partner systems with an equal or lower security level.

100 enables the basic function *inbound follow-up processing + preprocessing + postprocessing* for all partner systems.

-ip not specified
leaves the setting for *inbound follow-up processing + preprocessing + postprocessing* unchanged.

-if=s | -if=0..100
sets the value for the basic function *inbound file management*.

s sets the value defined in the standard admission set.

0 disables the basic function *inbound file management*.

1 .. 99
The basic function *inbound file management* is released only for partner systems with an equal or lower security level.

100 enables the basic function *inbound file management* for all partner systems.

Please note that the subcomponent "display file attributes" is controlled via the basic function *send inbound*. Some subcomponents affect other settings (see the following table):

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled
Modify file attributes	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Rename files	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Delete files	Inbound Receive(IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management(IBF) enabled
Create, rename and delete directories	Inbound File Management(IBF) enabled **and** direction= from partner in profile

-if not specified
leaves the setting for *inbound file management* unchanged.

5.17 ftmodi - Modify an instance

The *ftmodi* command allows you to assign another Internet host name address to an instance.

Format

ftmodi -h |
<instance 1..8>
[-addr=<host name> | -addr=@n]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance
Name of the instance to be modified.
Instance names have a maximum length of 8 characters and must consist of alphanumeric characters. The first character must not be a number.

-addr= host name | **-addr**=**@n**
Internet host name by which the instance is addressed.

host name
Another Internet host name can be assigned to the instance here.

@n for *host name*
This specification is only permitted for the standard instance *std*.

The standard instance is not assigned a host address anymore, and therefore it signs on for all addresses of the system.

In this manner you can switch from an operation with several instances to a one instance operation.

Messages of the ftmodi command

If *ftmodi* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

5.18 ftmodo - Modify operating parameters

You can use *ftmodo* to modify the following parameters for *open*FT operation:

– the key length of the RSA key
– the maximum values for file transfer
– the identification and the name of the local system
– the default value for the security level
– the mode for sender verification
– the logging scope (traces, loggingand console traps)
– the variant of the used code table
– the port number for the individual protocols

For FTAM operation, you can also activate or deactivate the Application Entity Title (AET).

Format

```
ftmodo  -h  |
    [ -kl=0 | -kl=768 | -kl=1024]
    [ -tu=<transport unit size 512..65535>]
    [ -pl=1 | -pl=]
    [ -cl=<connection imit 1..255>]
    [ -rql=<maximum number of requests 2..32000>]
    [ -rqt=<request lifetime 1..400> | -rqt=]
    [ -id=<identification 1..64>]
    [ -p=<prozessor name 1..8>] [ -l=<station name 1..8>]
    [ -sl=<security level 1..100> | -sl=p]  [ -ptc=i | -ptc=a]
    [ -lt=a | lt=f | lt=n] [ -lc=a | -lc=m | -lc=r]
    [ -tr=n | -tr=f | -tr=c]
    [ -trp=a | -trp=[openft] [,][ftam] [,][ftp]]
    [ -trr=[l | r] [a | s]] [ -tro=[b]]
    [ -tpc=a | -tpc=n | -tpc=[[-]sss], [[-]fts],
                        [[-]rqs], [[-]rqc], [[-]rqf], [[-]pts], [[-]ptu]]
    [ -ccs=<CCS name 1..8>]
    [ -ftp=<port number 1..65535> | -ftp=@s | -ftp=]
    [ -openft=[<port number1..65535>][.<T-Sel 1..8>]  |  -openft=@s]
    [ -ftam=[<port number 1..65535>][.<T-Sel>[.<S-Sel>[.<P-Sel>]]] |
      -ftam=@s]
    [ -ftstd=<port number 1..65535> | -ftstd=@s]
    [ -tns=y | -tns=n]
    [ -ae=y | -ae=n]
    [ -dp=n | -dp=f]
```

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-kl=0 | **-kl=**768 | **-kl=**1024

The *-kl* parameter can be used to change the length of the RSA key used in encryption. The value of the *kl* parameter specifies the new RSA key length in bits. The RSA key is only used for the encryption of the AES key agreed between the partners (or for encrypting the DES key in versions up to *open*FT V7.0).
*open*FT uses the AES key for encrypting request description data and any file content present.

The *ftmodo -kl=...* command can be specified in current *open*FT operation.

-kl=0 explicitly deactivates encryption. If this is set during operation then any requests with encryption (prior to *ftmodo -kl=0*) that have been submitted but not yet started are aborted with errors. Any running requests are processed and their encryption is retained. New requests are rejected.

After reinstallation, the default value *-kl=768* is used.

-tu=transport unit size

You use the parameter *-tu* to define the upper limit for message length at transport level (block length). You can choose a value between 512 and 65535.

The default value is 65535 characters.

The block length only applies to requests to *open*FT partners.

-pl=1 | **-pl=**

Maximum number of processes used for the processing of asynchronous requests.

1 All asynchronous requests are processed by the same process.

No value specified (default value)

If you specify *-pl=* without parameters then the number of processes is equal to the number of connections, i.e. each connection is handled by a separate process.

-cl=connection limit

Maximum number of asynchronous requests that are processed simultaneously.

The default value is 16.

-rql= maximum number of requests

You use *-rql* to specify the maximum number of entries in the request queue. You can choose a value between 2 and 32000.

The default value is 2000.

-rqt= request lifetime | **-rqt**=

You use *-rqt* to specify the maximum lifetime of requests in the request queue. The value applies to both inbound and outbound requests and is specified in days. Values between 1 and 400 are permitted. Once the specified period has expired, requests are deleted from the request queue.

The default value is 30 days.

request lifetime not specified:

If you specify *-rqt*= without parameters then the maximum lifetime is reset to the default value.

-id=identification

Specifying the instance identification of your instance. Partner systems using *open*FT Version 8.1 and later, address your system via this string. In return, *open*FT uses the instance ID as the sender address when addressing the partners. The instance ID must be unique and not case-sensitive (see also section “Instance Identifications” on page 27). If you modify the instance ID, the relevant public key files will be automatically updated.

-p=processor name

You specify the processor name assigned to your system here.

-l=station name

The station name of the *open*FT application. The default value is $FJAM.

The specifications for *processor name* and *station name* depend on how your system is connected to the network. Further details can be found in the chapter “Installation and configuration” on page 43.

-sl=security level | **-sl**=p

You use this option to define the default security level. This level applies to partners in the partner list to which no explicit security level value has been assigned as well as to partners which are not entered in the partner list. The effect also depends on the settings for the admission set, see the *ftmoda* command on page 100.

security level

Specifies a fixed default security level. Values between 1 and 100 are permitted. 1 indicates a very low and 100 a very high requirement for protection with regard to the partners.

p The default security level depends on the partner's attributes:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system.
- Security level 100 otherwise, i.e. if the partner has only been identified by its address.

The default value is *-sl=p*.

-ptc=i | **-ptc**=a

This allows you to modify the global settings for sender verification. This setting only applies to partners which are connected via the *open*FT protocol and do not use authentication (e.g. partners with *open*FT V8.0 or earlier).

i (identification)

Deactivates verification of the transport address. Only the identification of the partner is checked.

a (address)

Activates verification of the transport address
If the transport address under which the partner logs in does not correspond to the entry in the partner list then the request is rejected.

-lt=a | **-lt**=f | **-lt**=n

This option is used to selectively deactivate FT log records.
With connections to FTAM partners, it can take up to a minute for the selection to become active.

a (all)

This is the default setting; log records are written for all FT requests.

f (failure case)

Log records are written for failed FT requests only.

n (none)

No log records are written.

-lc=a | **-lc**=m | **-lc**=r

This option is used to selectively activate/deactivate FTAC log records.
With connections to FTAM partners, it can take up to a minute for the selection to become active.

a (all)

This is the default setting; log records are written for all FTAC access checks.

m (modifying FM calls)

Log records are written for all modifying file management requests leaving the remote system as well as for all rejected FTAC access checks.

r (reject case)

Log records are written for rejected FTAC access checks only

-tr=n | **-tr**=f | **-tr**=c

This allows you to activate and deactivate the *open*FT monitoring function

n (on)

The *open*FT monitoring function is activated.

f (off)

The *open*FT monitoring function is deactivated.

c (change)

The current monitoring file is closed and a new one is opened.

-trp=a | **-trp**=[openft] [,][ftam] [,][ftp]]

This option allows you to choose partners on a protocol-specific basis by specifying a comma-separated list of one or more protocol types. All the partners that are addressed via this or these protocol type(s) are then monitored. You can modify the selection made here on a partner-specific basis, see the *–tr* option in the *ftmodptn* command on page 126.

a (all)
All protocol types, and consequently all partners, are selected for monitoring.

openft
All partners addressed via the openFT protocol are selected for monitoring.

ftam All partners addressed via the FTAM protocol are selected for monitoring.

ftp All partners addressed via the ftp protocol are selected for monitoring.

No protocol type selected
If you specify *-trp=* without parameters then no partner is selected for monitoring. In this case, only those partners for which monitoring has been activated on a partner-specific basis using *ftmodptn ... tr=n* are monitored, see page 126.

-trr=[l | r] [a | s]

This option allows you to select the request types that are to be monitored.

l (local)
All locally submitted requests are selected for monitoring.

r (remote)
All remotely submitted requests are selected for monitoring.

a (asynchronous)
All asynchronous requests are selected for monitoring.

s (synchronous)
All synchronous requests are selected for monitoring.

No request type specified
If you specify *-trr=* without parameters then all requests are selected for monitoring.

-tro=[b]

You can use *-tro* to select options for the monitoring function. These options are only effective if the monitoring function is active.

b (no bulk data)

Minimum trace. Only protocol elements with no file contents (bulk data) are written to the monitoring file. In the case of protocol elements with file contents, the monitoring file simply notes that records have been suppressed at this point. This note is entered only once for a sequence of similar records.

This option is only available for *open*FT and FTP partner.

No option specified

If you specify *-tro=* without parameters then the trace is written normally.

-tpc=a | **-tpc**=n | Console trap list (comma-separated)

You use -tpc to activate and deactivate console traps.

In UNIX systems and Windows systems, console traps are written to the *open*FT file *conslog*. In UNIX systems, BS2000 and z/OS they are also output at the console and in Windows systems they are also written to the event log.

For *-tpc* you can enter the following values:

a (all)

All traps are written.

n (none)

No traps are written.

sss Activates traps relating to the status of the *open*FT subsystem.

-sss Deactivates traps relating to the status of the *open*FT subsystem.

fts Activates traps relating to the status of the asynchronous server.

-fts Deactivates traps relating to the status of the asynchronous server.

rqs Activates traps relating to the status of the request queue.

-rqs Deactivates traps relating to the status of the request queue.

rqc Activates traps on the successful termination of a request.

-rqc Deactivates traps on the successful termination of a request.

rqf Activates traps on the unsuccessful termination of a request.

-rqf Deactivates traps on the unsuccessful termination of a request.

pts Activates traps relating to the status of partner systems.

-pts Deactivates traps relating to the status of partner systems.

ptu Activates traps when a partner system is inaccessible.

-ptu Deactivates traps when a partner system is inaccessible.

-ccs=CCS name

You use *CCS name* to define a new character set which is represented by a code table. This character set is then used as the new default value for transfer requests (*ft, ncopy*). The code table specification is only relevant for requests to *open*FT partners.

Default value: iso88591 (corresponds to ISO8859-1)

Another character set can be explicitly assigned for *ft* and *ncopy* (options *-lc* and *-rc*).

You can also define your own character set. For details concerning CCS names and the associated code tables, see section “Administering code tables” on page 18.

-ftp=port number | **-ftp**=@s | **-ftp**=

You use *port number* to specify the port number used by FTP.
Possible values: 1 to 65535.

The default number is 21.

@s Sets the port number for FTP to the default value of 21.

port number not specified

You use *-ftp*= to set the FTP server to inactive, i.e. it cannot accept any more inbound requests.

-openft=port number[.T-selector] | **-openft**=@s

For operating without TNS, you can use *port number* to specify a port number other than the default for the local *open*FT server. You can also specify a T-selector of between 1 and 8 characters in length. In this case, the port number and T-selector must be separated by a period.
Possible values for the port number: 1 to 65535
The default value for the port number is 1100.

For operating with TNS, you can specify a TNS name other than the default for the local *open*FT server. A period must be placed before the TNS name, e.g. *-openft=.OPNFTSRV.* The TNS name must not contain any period.
The default value for the TNS name is $FJAM.

Please use this function carefully because setting a port number or TNS name other than the default makes it difficult for *open*FT partners to address the local system!

@s *-openft=@s* sets the port number and the TNS name for the *open*FT server to the default value, i.e.:
1100 (port number for operating without TNS) and
$FJAM (TNS name for operating with TNS)

-ftam=port number[.T-selector[.S-selector[.P-selector]]] | **-ftam**=@s
For operating without TNS, you can use *port number* to specify a port number other than the default for the local FTAM server. You can also specify a T-selector, a session selector and a presentation selector, each of which may have a length of 1 to 16 characters. In this case, the port number, T-selector, S-selector and P-selector must be separated by a period.
Possible values for the port number: 1 to 65535
The default value for the port number is 4800.

For operating with TNS, you can specify a TNS name other than the default for the local FTAM server. A period must be placed before the TNS name, e.g. *-ftam=.FTAMSERV.* The TNS name must not contain any period.
The default value for the TNS name is $FTAM.

Please use this function carefully because setting a port number or TNS name other than the default makes it difficult for FTAM partners to address the local system!

@s *-ftam=@s* sets the port number and the TNS name for the FTAM server to the default value, i.e.:
4800 (port number for operating without TNS) and
$FTAM (TNS name for operating with TNS)

-ftstd=port number | **-ftstd**=@s
You use *port number* to define a port number other than the default for the addressing of *open*FT partners via their host names.
Possible values: 1 to 65535

The default value is 1100.

Please use this function carefully because setting a port number other than the default makes it impossible to access *open*FT partners which use the default port number and are addressed via their host names!

@s *-ftstd=@s* deletes a port number other then the default for the addressing of *open*FT partners via their host names. The default port number of 1100 then applies again.

-tns=y | **-tns**=n

This option allows you to activate or deactivate the use of TNS names. This does not affect the use of TCP/IP host names, IP addresses or partner management.

y This activates the use of TNS names for *open*FT and FTAM transfer.

This is necessary, for example, if other transport protocols are to be used alongside TCP/IP.

n This deactivates the use of TNS names. In this case, it is only possible to use the TCP/IP transport protocol. By default, the port numbers set in the operating parameters are used for communications (options *-openft*, *-ftam* and *-ftstd*).

-ae=y | **-ae**=n

This option activates/deactivates the AET (Application Entity Title).

y A "nil Application Entity Title" is included as the calling or called Application Entity Title (AET) for transfer using the FTAM protocol.

n The AET is deactivated. The option only has to be reset to *-ae=n* if FTAM link partners, as responders, do not expect to receive an AET.

-dp=n | **-dp**=f

This option allows you to activate or deactivate the dynamic entries in the partner list.

n (on) This activates the dynamic partner entries. Partners can then be accessed via their address even if they are not entered in the partner list

f (off) This deactivates the dynamic partner entries, i.e. partners cannot be accessed via their address. As a result, it is only possible to use partners that are entered in the partner list and are addressed via the partner name.

Examples

1. The identification of your own instance is to be set to host.hugo.net:

   ```
   ftmodo -id=host.hugo.net
   ```

2. Only partners from the partner list are to be permitted:

   ```
   ftmodo -dp=f
   ```

5.19 ftmodp - Modify FT profiles

ftmodp stands for "modify profile". The FTAC administrator can use this command to change or to privilege FT profiles of other users.

In the event that the FTAC administrator does not have any root admission, then admission profiles of other users are blocked after a modification (except after *-priv=y*). This can be by-passed by entering *-ua=user ID,password*. If the user later changes his/her password, the profile will no longer be usable without further modification.

Format

```
ftmodp -h |
          <profile name 1..8> | @a
        [ -s=[<transfer admission 8..32> | @a | @n ][,<user ID 1..32> | @a]]
        [ -ua=[ <user ID 1..32> ],[<password 1..20> | @n ] ]
        [ -nn=<profile name 1..8> ]
        [ -tad= | -tad=<transfer admission 8..32> | -tad=@n ]
        [ -v=y | -v=n ]     [ -d=[yyyymmdd | -d=]
        [ -u=pr | -u=pu ]   [ -priv=y | -priv=n ]
        [ -iml=y | -iml=n ]
        [ -iis=y | -iis=n ]   [ -iir=y | -iir=n ]
        [ -iip=y | -iip=n ]   [ -iif=y | -iif=n ]
        [ -ff= | -ff=t | -ff=m | -ff=r | -ff=p | -ff=tmrp | -ff=prmt ]
        [ -dir=f | -dir=t | -dir=ft ]
        [ -pn=<partner 1..200>,...,<partner(50) 1..200> | -pn=]
        [ -pna=<partner 1..200>,...,<partner(50) 1..200> ]
        [ -pnr=<partner 1..200>,...,<partner(50) 1..200> ]
        [ -fn=<file name 1..512> | -fn=] [ -fnp=<file name prefix 1..511>]
        [ -ls= | -ls=@n | -ls=<command1 1..1000> ]
        [ -lsp= | -lsp=[<command2 1..999>] [ -lss= | -lss=command3 1..999>]
        [ -lf= | -lf=@n | -lf=<command4 1..1000> | ]
        [ -lfp= | -lfp=<command5 1..999>] [-lfs= | -lfs=<command6 1..999>]
        [ -wm=o | -wm=n | -wm=e | -wm=one ]
        [ -c=y | -c=n]
        [ -txt=<text 1..100> | -txt=]
```

Description

In the following, only those options and values which are particularly important for the administrator or which offer the administrator additional functionality are described in detail. The remaining options are described in the User Guide.

profile name
: specifies the name of the FT profile you wish to modify. To see the profile names you have already assigned, you can issue the *ftshwp* command (without options).

  @a for profile name
  : modifies all FT profiles that come into question at once, unless you select a specific profile with the option *-s*.

-s=[transfer admission | @n | @a] [,user ID | @a]
: is used to specify selection criteria for the FT profile to be modified.

  transfer admission
  : specifies the transfer admission of the FT profile to be modified. You must specify a binary transfer admission in the form x\'...\' or X\'...\'.

  @a for *transfer admission*
  : modifies either the FT profile specified with *profile name* (see above) or (if no profile name was specified) all the profiles that come into question.

  @n for *transfer admission*
  : selects all FT profiles without transfer admission.

  transfer admission not specified
  : causes to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time. If you just press ↵, this has the same effect as specifying *@a*.

  ,user ID
  : As the FTAC administrator, you can specify any login name here.

  @a for *user ID*
  : If you specify *@a* as the FTAC administrator, you can modify the FT profiles for any login names.

user ID not specified
: modifies only profiles belonging to the user's own login name, regardless of who issues the command.

-s not specified
: if *@a* is specified for *profile name*, all the FT profiles belonging to the login name under which the *ftmodp* command is issued are modified. Otherwise, the FT profile with the specified name is modified.

-ua=[user ID],[password | @n]
: With *-ua*, the FTAC administrator can assign any desired FT profile of a login name to another login name.

  user ID
  : As the FTAC administrator, you can specify any login name here.

  ,password
  : specifies the password for a login name. A binary password must be specified in the form x\'...\' or X\'...\'. The FT profile for the login name is valid only so long as the password *password* is valid for the login name. When the password is changed, the profile can no longer be used (not locked!).

  @n for *password*
  : In this case, the FTAC administrator cannot specify any transfer admission for the FT profile if you do not have root privileges (UID=0). An existing transfer admission will be automatically deleted in this case.

  comma only (,) no *password* specified
  : causes FTAC to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. In this case, single quotes must not be escaped by a backslash. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time.

  user ID only (without comma and *password*) specified
  : means that the profile is valid again for all passwords of the specified login name *user ID*.

-ua␣not specified
: the login name of this FT profile remains unchanged.

-tad=[transfer admission | @n]

allows you to modify the transfer admission of an FT profile. As the FTAC administrator, you can also modify the transfer admissions for other login names if you have root privileges (UID=0).

transfer admission

The transfer admission must be unique within your UNIX system so that there are no conflicts with transfer admissions defined by other FTAC users for other access permissions. A binary transfer admission must be specified in hexadecimal format in the form x\'...\' or X\'...\'. If the transfer admission you select has already been assigned, FTAC rejects the *ftmodp* command and issues the message

```
Transfer admission already exists.
```

@n for *transfer admission*

disables the old transfer admission.

transfer admission not specified

-tad= causes FTAC to prompt you to enter the transfer admission after the command has been entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program expects you to enter the transfer admission a second time as an entry check.

-tad not specified

does not modify the transfer admission of the FT profile.

-priv=y | -priv=n

This option is used by the FTAC administrator to grant privileged status to an FT profile.

y grants privileged status to the FT profile. The FT administrator's entries in the admission set are ignored for requests executed with a privileged FT profile, i.e., if the user uses the *-iml, -iis, -iir, -iip* or *-iif* options in the FT profile, both the user' s entries (MAX. USER LEVELS) and the administrator' s entries (MAX. ADM LEVELS) are ignored.

n withdraws the privileged status, if it had been granted, from the FT profile.

-priv not specified

does not modify the privileged status of the FT profile.

-iml=y | -iml=n
-iis=y | -iis=n
-iir=y | -iir=n
-iip=y | -iip=n
-iif=y | -iif=n

These options are used to specify whether the FT profile is to be restricted by the values in the admission set (MAX. USER LEVELS). If the FT profile is also privileged by you as the FTAC administrator, the entries you have made (the MAX. ADM LEVELS) can also be ignored. This FT profile would then allow *inbound* basic functions which are disabled in the admission set to be used. Possible values are:

y allows the values in the admission set to be ignored.

n (default value)
restricts the functionality of the profile to the values in the admission set.

-ixx not specified
The existing definitions of the profile for the basic functions involved remain in effect.

The following table shows which subcomponents of the file management can be used under which conditions.

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled
Modify file attributes	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Rename files	Inbound Receive(IBR) **and** Inbound File Management(IBF) enabled
Delete files	Inbound Receive(IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management(IBF) enabled
Create, rename and delete directories	Inbound File Management(IBF) enabled **and** direction= from partner in profile

5.20 ftmodptn - Modify partner properties

You use the *ftmodptn* command to modify the properties of partner systems in the local system's partner list.

Format

ftmodptn -h |
<partner 1..200> | @a
[-pa=<partner address 1..200>]
[-id=<identification 1..64> | -id=]
[-ri=<routing info 1..8> | -ri=@i | -ri=]
[-ptc=i | -ptc=a | -ptc=]
[-sl=1..100 | -sl=p | -sl=]
[-st=a | -st=d | -st=ad]
[-am=n | -am=y]
[-tr=n | -tr=f | -tr=]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner | **@a**

partner is the name of the partner system in the partner list or the address of the partner system whose properties you want to modify.

@a for *partner*

Partner is not a selection criterion, i.e. you modify the properties of all the partner systems present in the partner list. This specification is only of use in combination with the options *-ptc*, *-sl*, *-st*, *-tr* and, under certain circumstances, *-ri*.

-pa=partner address

[protocol://]host[:[port].[tsel].[ssel].[psel]]

protocol

Protocol stack via which the partner is addressed.
Possible values:

openft

openFT partner, i.e. communication via the *open*FT protocol

ftam FTAM partner, i.e. communication via the FTAM protocol

ftp ftp partner, i.e. communication via the ftp protocol

Default value: **openft**

Exception: if a global name from the TNS is used for *host* and a presentation selector is assigned to this global name then *ftam* is the default value. Please note that TNS use must be activated for this (*ftmodo -tns=y*).

host Internet host name, IP address or GLOBAL NAME from the TNS, mandatory parameter. Format of IP addresses (example):
`%ip111.222.123.234` (ipv4) or
`%ip6[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]` (ipv6) or
`%ip6[FE80::20C:29ff:fe22:b670%5]` (ipv6 with Scope-ID)

The brackets [..] are mandatory with ipv6.

port Port number in the case of a TCP/IP connection, optional.

tsel Transport selector (only *open*FT and FTAM protocol), optional.

ssel Session selector for FTAM connection or session routing information in the case of *open*FT protocol with *open*FTIF connection, optional.

psel Presentation selector for FTAM connection, optional.

For details concerning address specifications, see *open*FT User Guide.

-*pa* not specified
The partner address is unchanged.

-id=identification | **-id**=
Identification unique in the network of the *open*FT instance in the partner system. In the case of FTAM partners, it is possible to specify an Application Entity Title in the form n1.n2.n3.n4..mmm as the identification. n1, n2 etc. are positive integer values which describe the "Application Process Title". n1 can only have the values 0, 1 or 2, n2 is restricted to values between 0 and 39 if n1 does not have the value 2. The optional Application Entity Qualifier mmm must be separated from the values of the Application Process Title by two periods. For details, see the *open*FT User Guide.

-*id* not specified or *id*=
The setting for identification is unchanged.

-ri=routing info | **-ri=@i** | **-ri=**

If the partner system can only be accessed via an intermediate instance (e.g. *open*FTIF gateway) then you specify the address information to be used for routing by the intermediate instance in *routing info*.

@i for *routing info*

The instance identification specified in *-id=* is used as the routing information.

neither *@i* nor *routing info* specified

The specification of *-ri=* (without parameters) means that the partner system can be accessed directly, i.e. without an intermediate instance.

-ri not specified

The setting for the routing information is unchanged.

-ptc=i | **-ptc=a** | **-ptc=**

You can use *-ptc* to modify the operating parameter setting for sender verification on a partner-specific basis. These settings only affect partners which are connected via the *open*FT protocol and do not operate with authentication (e.g. partners with *open*FT V8.0 or earlier).

i (identification)

Deactivates checking of the transport address. Only the partner's identification is checked. The partner's transport address is also not checked even if extended sender verification is globally active (see the *ftmodo* command on page 106).

a (address)

Activates checking of the transport address. The partner's transport address is checked even if checking of the transport address is globally deactivated (see *ftmodo* command on page 106).

If the transport address under which the partner logs on is not the same as the entry in the partner list then the request is rejected.

neither *i* nor *a* specified

-ptc= (without parameters) means that the operating system parameters apply to sender verification.

-ptc not specified

The setting for sender verification is unchanged.

-sl=1..100 | **-sl**=p | **-sl**=

You use this option to assign a security level to the specified partner system or to all the partner systems.

1..100

Specifies a fixed security level. 1 is the lowest and 100 the highest security level.

p Assigns a security level to the partner depending on the partner's attributes, i.e.:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system.
- Security level 100 otherwise, i.e. if the partner has only been identified by its address.

security level not specified

-sl= (without parameters) means that the operating parameter setting for the security level applies (see command *ftmodo* on page 106)

-*sl* not specified

The setting for the security level is unchanged.

-st=a | **-st=**d | **-st=**ad

This option allows you to control how locally submitted asynchronous file transfer requests to the specified partner system or systems are processed.

a (active)

Locally submitted asynchronous file transfer requests are processed if the asynchronous *open*FT server is started.

d (deactive)

Locally submitted asynchronous file transfer requests are initially not processed but are stored in the request queue.

ad (automatic deactivation)

Unsuccessful attempts to establish a connection to this partner system result in its deactivation. The maximum number of unsuccessful attempts is 5. If you want to perform file transfer again with this system, you must explicitly activate it with *ftmodptn -st=a*.

-*st* not specified

The processing mode is unchanged.

-am=n | **-am**=y
: You can use this option to force partner authentication.

  n Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

  y Authentication is forced, i.e. requests are only processed if the local system is successfully able to authenticate the partner, see page 27.

-am not specified
: The authentication mode is unchanged.

-tr=n | **-tr**=f | **-tr**=
: You can use this option to modify the operating parameter settings for the partner selection for the *open*FT monitoring function on a partner-specific basis.

  n (on)
  : The monitoring function is active for this partner or for all the partners. However, a trace is only written if the *open*FT monitoring function has been activated via the operating parameters, see page 106ff, *ftmodo*, option *-tr*.

  f (off)
  : The monitoring function is deactivated for this partner or for all partners.

  neither *n* nor *f* specified
  : *-tr*= (without parameters) means that the operating parameter setting for the partner selection in the *open*FT monitoring function applies (see the *ftmodo* command on page 106).

-tr not specified
: The setting for the monitoring function is unchanged.

5.21 ftmodr - Change the property of requests

With the *ftmodr* command, you can change the priority of requests you have issued, or of a group of requests, for example all the requests to a particular partner. Furthermore, you have the option of changing the order of requests within a priority.

As the FT administrator, you can change the priority of all requests in the system.

Format

ftmodr -h |
[-ua=<user ID 1..32> | -ua=@a]
[-pn=<partner 1..200>]
[-fn=<file name 1..512|]
[-pr=n | -pr=l] [-qp=f | -qp=l]
[<request ID 1..2147483647>]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ua=user ID | **-ua=@a**
You use *-ua* to specify the user ID for which requests are to be modified.

user ID
As FT administrator, you may specify any user ID here.

@a As FT administrator, you can specify *@a* to modify requests relating to all user IDs.

-ua= not specified
Your own user ID is the selection criterion. Exception: you called the command as FT administrator and also specified a request ID: in this case, the presetting is *@a*.

-pn=partner
You use *-pn* to specify a name or an address for the partner system for which you want to modify requests. The partner should be specified in the same way as in the request or as it is output in the *ftshwr* command without the option *-s*, *-l* or *-csv*. If *open*FT finds a partner in the partner list

that corresponds to the specified partner address then *ftshwr* indicates the name of the partner even if a partner address was specified on request entry.

-fn=file name
: You use *-fn* to specify the file name for which requests are to be modified. Requests which access this file in the local system are modified.

  You must specify the file name that was used when the request was created. This file name is also output by the *ftshwr* command without the *-fn* option.

  Wildcards may not be used in the file name.

-pr=n | **-pr**=l
: indicates the new priority. The following values are possible:

  n (normal)
  : the request has the priority "normal"

  l (low)
  : the request has the priority "low"

-qp=f | **-qp**=l
: indicates the position of the request within the same priority. The following values are possible:

  f (first)
  : the request is placed at the top of the list of requests with the same priority

  l (last)
  : the request is placed at the bottom of the list of requests with the same priority.

-id=request ID
: *request ID* is used to specify the identification of a specific request that is to be modified. The request ID is output on the screen when reception of the request is confirmed. It can also be displayed using the *ftshwr* command.

If you have specified a request ID but the other specified selection criteria do not match the request then the request is not modified and the following error message is output:

`ftmodr: Request` ***request ID*** `not found`

5.22 ftremptn - Remove a partner from the partner list

Format

ftremptn [-h] |
<partner 1..200>

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner
Specifies the partner that is to be removed from the partner list. You can specify the name in the partner list or the partner's address. The name and address are displayed using the *ftshwptn* command.

All requests stored for this partner in the request queue are deleted. This is even the case for requests with a status which means that they are known to the partner system. Since this can lead to inconsistencies, you should only remove a partner from the partner list if either there are no more requests for this partner in the request queue or if you can be sure that the partner system will not become active again.

5.23 ftsetjava - Manage link to the Java executable

ftsetjava is used to set a link to the Java executable.

ftsetjava is used implicitly during installation of *open*FT. In addition, you can also call *ftsetjava* as administrator in order to

– see what file is referenced by the link to the Java executable used by *open*FT

– set the link if Java was not installed or if an incorrect version was installed at the time when *open*FT was installed or if the installation path of the Java executable has changed.

– see what Java installations are present in the directories searched by *open*FT.

Format

ftsetjava [@s | @a]

Description

@s Sets the link to the Java executable.
If the attempt to set a link to the Java executable fails because no suitable Java installation is available, an appropriate message is output to *stdout*. A warning is also issued if this happens during installation of *open*FT.

@a Shows all the Java executables installed in the search path. Any subsequent call to *ftsetjava @s* is successful if and only if at least one of these installations meets the requirements of *open*FT with respect to the version. The first file with a suitable version in the list of Java executables which is output is then used as the source of the link.

neither *@s* nor *@a* specified
If *ftsetjava* is called without parameters, it outputs the Java file used by *open*FT.

5.24 ftshwa - Display admission sets

ftshwa stands for "show admission set", and allows you to examine admission sets.

As the FTAC administrator, you can obtain information on all admission sets in your system.

It outputs the following information:

- what limit values the owner of the user ID has set for the individual basic functions
- what limit values the FTAC administrator has set for the user ID for the individual basic functions,
- whether or not the admission set is privileged (i.e. who is the FTAC administrator).

Format

ftshwa -h |
[<user ID 1..32> | @a | @s] [-csv]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

user ID | @a | @s
specifies the user id of for which the admission set is to be displayed.

user ID
As the FTAC administrator, you can specify any login name desired.

If a login name longer than 8 characters is specified, the first 7 characters are output followed by an asterisk (*).

@a for *user ID*
When entered by the FTAC administrator, *@a* displays information on the standard admission set and all admission sets that differ from it.

@s for *user ID*
returns information only on the standard admission set.

If you specify a non-existent login name for *user ID*, the current standard admission set is displayed.

user ID not specified
FTAC displays information on the admission set of the login name under which *ftshwa* was entered.

-csv Specifying *-csv* indicates that the FT admission sets are to be output in the CSV format. The values in the output are separated by semicolons.

-csv not specified
The FT admission sets are output in the standard format.

Example

Display of command `ftshwa␣@a`:

```
$ ftshwa @a

               MAX. USER LEVELS              MAX. ADM LEVELS          ATTR
USER-ID  OBS  OBR  IBS  IBR  IBP  IBF  OBS  OBR  IBS  IBR  IBP  IBF
*STD     100  100  100  100  100  100  100  100  100  100  100  100
john     100* 100* 100* 100* 100* 100* 100* 100* 100* 100* 100* 100*
 root      50   50    1    1    1    1  100* 100* 100* 100* 100* 100* PRIV
 smith     90   90    0    0    0   90  100* 100* 100* 100* 100* 100*
```

The displayed information has the following meaning:

USER-ID

The USER-ID column contains the login names to which the respective admission sets belong. If a login name longer than 8 characters is specified, the first 7 characters are output followed by an asterisk (*).

MAX-USER-LEVELS / MAX-ADM-LEVELS

The six columns under MAX-USER-LEVELS show the values specified by each of these FTAC users for their respective admission sets. The six columns under MAX-ADM-LEVELS contain the values set by the FTAC administrator. The lower of the two values determines whether or not the owner of this admission set may use the basic function specified.

The names of the basic functions are abbreviated as follows:
OBS = **O**UT**B**OUND-**S**END
OBR = **O**UT**B**OUND-**R**ECEIVE
IBS = **I**N**B**OUND-**S**END
IBR = **I**N**B**OUND-**R**ECEIVE
IBP = **I**N**B**OUND-**P**ROCESSING
IBF = **I**N**B**OUND-**F**ILE-MANAGEMENT

The values in the admission set have the following meaning:

0	The basic function is disabled.
1..99	The basic function is only released for partner systems with the same or a lower security level. You can use the *ftshwptn* command to display a partner system's security level.
100	The inbound basic function is enabled for all partner systems.

An asterisk '*' after the value indicates that this entry was taken from the standard admission set and will automatically be modified if the value in the standard admission set is changed.

ATTR

PRIV in the ATTR column indicates the privileged admission set; *root* is the FTAC administrator. At the present time, there are no further attributes for the ATTR column.

5.25 ftshwd - Display diagnostic information

With the *ftshwd* command, you can display diagnostic information.

The diagnostic documents are used by the Maintenance and Diagnostic Service of Fujitsu Siemens Computers for error diagnosis.

Format

ftshwd

Description

The command has no further switches. The following example shows the output for this command, and explains the meanings of the fields.

```
DATE     TIME    SSID  COMPONENT    LOCATION-ID       INFO
20060411 100921  FT    251/yfysequ  46/SwinsLwrite    ffffffff
20060411 100923  FTAC  39/yfslogg   1/WriteErr        ffffffff
```

DATE
: Date when the error occurred

TIME
: Time at which the error occurred

SSID
: Subsystem ID; possible values: FT/FTAC

COMPONENT
: Module number/name

LOCATION-ID
: Function number/name

INFO
: Error code

5.26 ftshwe - Display FT profiles and admission sets from a file

ftshwe stands for "show environment", i.e. display FT profiles and admission sets from a file. Using *ftshwe*, the FTAC administrator can display FT profiles and admission sets that were saved using the *ftexpe* command.

Format

ftshwe -h |
<file name 1..512>
[-u=<user ID 1..32>[,...,<user ID(100) 1..32>]]
[-pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n]
[-as=y | -as=n]
[-l] [-csv]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name
file name specifies the file from which the FT profiles and admission sets are to be displayed.

-u=user ID1[,user ID2][,user ID3]..
specifies the user IDs whose FT profiles and admission sets are to be displayed. You can specify up to 100 login names simultaneously.

If the specified user ID has no admission sets, only the standard admission set is displayed.

If you specify a non-existent login name for *user ID1*, the current standard admission set is displayed.

-u not specified
all FT profiles and admission sets are displayed.

-pr=profile name1[,profile name2][,profile name3]... | -pr=@n
specifies the FT profiles to be displayed (up to 100).

@n for *profile name*
no FT profiles are displayed.

-pr not specified
: all FT profiles belonging to the user IDs specified in the -u parameter are displayed.

-as=y | -as=n
: specifies whether or not admission sets are to be displayed.

  y (default value)
  : all admission sets belonging to the login names specified in the -u parameter are displayed.

  n no admission sets are displayed.

-l specifies that you wish to see the contents of the selected FT profiles.

-l not specified
: displays only the names of the FT profiles. Markings also indicate whether or not an FT profile is privileged (*) and whether or not it is disabled (!).

-csv *-csv* specifies that the FT profiles and admission sets are to be output in CSV format. The values are output separated by semicolons. When -csv is specified, the output is always detailed (analogous to *-l*), regardless of whether or not -l is specified at the same time.

For details, see section “ftshwp” on page 185 and section “ftshwa” on page 179.

-csv not specified
: The FT profiles and admission sets are output in the standard format.

5.27 ftshwl - Display log records

With *ftshwl*, you can obtain information on all file transfer requests logged up to now by *open*FT.

As the administrator, you can display all log records in the system. The log records are stored in the file *inst_dir/instance/log/syslog*. Here, *instance* is the name and *inst-dir* the directory of the instance, see the *ftcrei* command on page 80.

The log records are marked as FT and FTAC log records respectively, which means that you can determine the type of log record from the output.

For every request, there is a FTAC log record in which you can find the result of the FTAC admission check. If the check is positive and *open*FT has accepted the request, there is also a second *open*FT log record which indicates whether the request was successfully executed or why it was aborted.

If no options are specified, *open*FT outputs the current log record. If options are specified, *open*FT outputs all log records up to the time specified in the command in reverse chronological order, i.e. starting from the most recent record to the oldest record.

There are three types of output: short output, long output and CSV output (**C**omma **S**eparated **V**alue).

Output is written to standard output.

Format

```
ftshwl  -h |
        [ <user ID 1..32> | @a]
        [ -rg=[[[[yyyy]mm]dd]hhmm|#1..99999999|0..999|:0..999]  [-
              [[[[yyyy]mm]dd]hhmm|#1..99999999|0..999|:0..999]]]
        [ -rt=t | -rt=c | -rt=tc | -rt=ct]
        [ -ff=[t][m][r][d][a][C][D][M]]
        [ -ini=l | -ini=r | -ini=lr | -ini=rl]
        [ -pn=<partner 1..200>]
        [ -fn=<file name 1..512>]
        [ -nb=1..99999999 | -nb=@a]
        [ -rc=0..ffff | -rc=@f]
        [ -l ] [ -csv ]
```

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

user ID | @a
: is used to specify the login name(s) for which log records are to be displayed. As the administrator, you can specify any login name.

  @a for *user ID*
  : FT or FTAC administrators can display the log records for all login names.

user ID not specified
: Only the log records for the login name under which the command was entered are displayed.

-rg=[[[[yyyy]mm]dd]hhmm]-[[[[yyyy]mm]dd]hhmm]
: You can *-rg* to specify the start and/or end of a logging interval.

  [[[yyyy]mm]dd]hhmm
  : When specifying a time, a 4-digit specification is interpreted as the time expressed in hours and minutes, a 6-digit specification as the day (date) and time in hours and minutes, an 8-digit specification as the month, day, and time in hours and minutes, and a 12-digit specification as the year, month, day, and time in hours and minutes. The largest possible value that can be specified as the date is 20380119 (January 19, 2038).
    *open*FT then displays all the log records written during the specified time period. The older time is taken to be the start time and the earlier time as the end time. This means that the time period is viewed from the past towards the present.
    The optional data ([…]) is automatically replaced by current values. If one of the limiting values is omitted, the current time is taken to be the end time, and the start time is the time at which the first log record was written.

-rg=[[[yyyy]mm]dd]hhmm
: If the minus sign is missing, the range is the exact minute specified. The largest possible value that can be specified as the date is 20380119 (January 19, 2038). The optional data ([…]) is automatically replaced by current values.

-rg=[#1..99999999]-[#1..99999999]

-rg is used to specify the start and/or end of a range of log IDs.

#1..99999999

The selection of a log ID is indicated by the leading # character. *open*FT then displays all the log records which lie within the specified range. The older time is taken to be the start and the earlier time as the end. This means that you are looking from the past towards the present with regard to the time and the log IDs.

If one of the limiting values is omitted, the current ID is used as the end log ID, and the ID of the first log written is used as the start log ID.

-rg=#1..99999999

If the minus sign is omitted, the range is restricted to the specified log ID only.

-rg=0..999 [**-**0..999]

Here you specify with *-rg* a relative time period as a multiple of 24 hours (i.e. as a number of days). You can specify a 1- to 3-digit number. *open*FT then outputs all the log records that are older than this.
You are thus looking backward in time.

-rg=:0..999 [-:0..999]

Here you specify with *-rg* a relative time period in minutes. You can specify a 1- to 3-digit number. You have the following options in this case:

- Specifying *-rg=:0..999 -:0..999* will output all log records that lie within the specified time range with respect to the current time.
- Specifying *-rg=:0..999* will output the log records in the time span that starts with the current time and goes back into the past by the number of minutes you have specified.
- Specifying *-rg=-:0..999* will output the log records that lie outside of the specified time limit, meaning all records that are older than the time specified in minutes.

You are thus looking back in time.

-rg not specified

The range is not a selection criterion.

-rt=t | **-rt**=c | **-rt**=tc | **-rt**=ct

Defines which type of log record is to be displayed.
Possible values are t, c, tc, or ct.

t The FT log records are displayed.
c The FTAC log records are displayed.
tc, ct The FT and FTAC log records are displayed.

-rt not specified
: The record type is not a selection criterion.

-ff=[t] [m] [r] [d] [a] [C] [D] [M]
: Defines the FT function for which log records are to be output. Possible values are: t, m, r, d, a, C, D and M or any combination of these values. The entries t, m, r, d, a, C, D and M are only valid for FTAC log records.

  t All log records for the function “transfer files” are output.

  m All log records for the function “modify file attributes” are output.

  r All log records for the function “read directories” are output.

  d All log records for the function “delete files” are output.

  a All log records for the function “read file attributes” are output.

  C All log records for the function “Create directory” are output.

  D All log records for the function “Delete directory” are output.

  M All log records for the function “Modify directory” are output.

-ff not specified
: The FT function is not a selection criterion.

-ini=l | **-ini**=r | **-ini**=lr | **-ini**= rl
: Defines the initiator for which log records are to be output. Possible values are: l, r, lr, rl.

  l Only log records belonging to file transfer functions issued locally are output.

  r Only log records belonging to file transfer and file management functions issued remotely are output.

  lr, rl The log records belonging to file transfer and file management functions issued locally and remotely are output.

-ini not specified
: The initiator is not a selection criterion.

-pn=partner
: Defines the partner system to which the log records are to be output. Partner is the name of the partner in the partner list or the address of the partner system.

-pn not specified
: The partner system is not a selection criterion.

-fn=file name
Defines the file to which the log records are to be output.

-fn not specified
The file name is not a selection criterion.

-nb=number | @a
Defines the number of log records to be output.

@a for *number*
All log records are output.

-nb not specified
If *-rg* has also been specified, *-nb* is replaced by the value *-nb=@a*.

If *-rg* is also not specified, *-nb* is replaced by the value *-nb=1*.

-rc=0..ffff | @f
Defines the reason code as a selection criterion for log record output.

0 .. ffff
All log records with a specified reason code are output.

@f All log records with reason codes other than 0000 are output. This criterion yields a list of log records for all requests terminated with error messages.

-rc not specified
The reason code is not a selection criterion.

-l Defines that the log records are to be output in long form.

-l not specified
The log records are output in short form.

-csv You can use *-csv* to specify that the log records are to be output in the CSV format. The values in the output are separated by semicolons. If *-csv* is specified, output is always in long form (analogous to *-l*) regardless of whether or not *-l* has also been specified.

-csv not specified
The log records are output in the standard format.

Examples

1. All log records that are more than two days (48 hours) old are output:

   ```
   ftshwl -rg=-2
   ```

2. All log records that are more than 15 minutes old but less than 30 minutes old are output:

   ```
   ftshwl  rg=:15-:30
   ```

3. All log records that are less than 30 minutes old are output:

   ```
   ftshwl -rg=:30
   ```

4. All log records that are more than 30 minutes old are output:

   ```
   ftshwl -rg=-:30
   ```

5.27.1 Description of log record output

Log records can be displayed using the graphical user interface or by using the *ftshwl* command. You can choose between a short overview, detailed information or, if further processing is to be performed with external programs, output in the CSV format.

The log records are identified by log IDs. The log IDs are assigned in ascending order, but for technical reasons the numbering of the log IDs is not always contiguous (i.e. there may be gaps).

The log record output and the reason codes of the logging function are described in the User Guide.

5.27.1.1 Logging requests with preprocessing/postprocessing

For security reasons, only the first 32 characters (or 42 characters in the case of *ftexecsv* preprocessing) of a preprocessing or postprocessing command are transferred to the log record. By arranging the call parameters appropriately or by inserting blanks, you can influence which command parameters do not appear in the log.

5.28 ftshwo - Display operating parameters

The *ftshwo* command outputs the operating parameters of the local *open*FT system. Alongside the standard output and output in CSV format, output may also be specified as a platform-specific command sequence. In this way, it is possible to save the settings and then load them onto another computer.

The FT administrator can set or modify the operating parameters with the *ftmodo* command.

Format

ftshwo -h |
[-csv | -px | -pw | -p2 | -pz]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-csv The operating parameters are output in CSV format. The individual values are separated by semicolons.

-px The operating parameters are output as a command string. This can be called as a shell procedure on UNIX systems in order to regenerate identical operating parameters on different systems.

-pw The operating parameters are output as a command string. This can be called as a batch procedure on Windows systems in order to regenerate identical operating parameters on different systems.

-p2 The operating parameters are output as a command string. This can be called as an SDF procedure on BS2000/OSD systems in order to regenerate identical operating parameters on different systems.

-pz The operating parameters are output as a command string. This can be called as a Clist procedure on z/OS systems in order to regenerate identical operating parameters on different systems.

No option specified
The operating parameters are output in standard format.

Output format of ftshwo

Example

```
# ftshwo
STARTED PROC-LIM CONN-LIM RQ-LIM MAX-RQ-LIFE TU-SIZE KEY-LEN CCS-NAME
  *YES     NONE       16     2000       30       32480    768   ISO88591
PTN-CHK DYN-PART SEC-LEV  FTAC-LOG FT-LOG
  STD       *ON        1       ALL     ALL
OPENFT-APPL       FTAM-APPL        FTP-PORT    USE TNS
*STD              *STD             4444          *NO
HOST-NAME         IDENTIFICATION / LOCAL SYSTEM NAME
ath00001          ath00001.city.cp.net / $FJAM,ATH00001

TRAP: SS-STATE FT-STATE PART-STATE PART-UNREA RQ-STATE TRANS-SUCC TRANS-FAIL
CONS     OFF       OFF        OFF         OFF        OFF        OFF         OFF

TRACE: SWITCH PARTNER-SELECTION    REQUEST-SELECTION    OPTIONS
        OFF   ALL                  ALL                  NONE
```

Meaning of the output together with the associated command options:

Field name	Meaning and values	Command/ option
STARTED	Specifies whether the asynchronous *open*FT server has started (*YES) or not (*NO).	*ftstart* *ftstop*
PROC-LIM	Maximum number of processes available for the processing of asynchronous requests.	*ftmodo -pl=*
CONN-LIM	Maximum number of asynchronous requests that can be processed simultaneously.	*ftmodo -cl=*
RQ-LIM	Maximum number of file transfer requests that can simultaneously be present in the local system's request queue.	*ftmodo -rql=*
MAX-RQ-LIFE	Maximum lifetime of requests in the request queue (in days).	*ftmodo -rqt=*
TU-SIZE	Upper limit for message length at transport level (in bytes)	*ftmodo -tu=*
KEY-LEN	Length of the RSA key currently used to encrypt the AES/DES key.	*ftmodo -kl=*
CCS-NAME	Name of the character set used by default for file transfer requests, see page 113	*ftmodo -ccs=*
PTN-CHK	Setting for sender verification: ADDR: address STD: identification	*ftmodo -ptc=*

Field name	Meaning and values	Command/ option
DYN-PART	Setting for dynamic partner entries: *ON (activated) *OFF (deactivated)	*ftmodo -dp=*
SEC-LEV	Default security level for partners in the partner list for which no security level has been set: 1..100: Fixed security level. 1 is the lowest and 100 the highest security level. B-P-ATTR: The security level is depending on the partner's attributes, i.e.: 10 if the partner has been authenticated. 90 if the partner is known in the transport system. 100 otherwise, i.e. if the partner has only been identified by its address.	*ftmodo -sl=*
FTAC-LOG	Scope of FTAC logging: ALL All FTAC access checks MODIFY Modifying file management requests and rejected FTAC access checks REJECTED Only rejected FTAC access checks	*ftmodo -lc=*
FT-LOG	Scope of FT logging: ALL: All requests FAIL: Only errored FT requests NONE: Logging deactivated	*ftmodo -lt=*
OPENFT-APPL	Port number of the local *open*FT server, possibly extended by the transport selector. STD means that the default value (1100) is used.	*ftmodo -openft=*
FTAM-APPL	Port number of the local FTAM server, possibly extended by the transport selector, the session selector and the presentation selector. STD means that the default value (4800) is used.	*ftmodo -ftam=*

Field name	Meaning and values	Command/ option
FTP-PORT	Port number used by FTP. NONE means that the FTP server is set to inactive.	*ftmodo -ftp=*
USE TNS	Specifies whether the TNS is to be used (*YES) or not (*NO).	*ftmodo -tns=*
HOST-NAME	Host name of the local computer, *NONE means that no host name has been assigned.	---
IDENTIFICATION	Instance identification of the local *open*FT instance.	*ftmodo -id=*
LOCAL-SYSTEM-NAME	Name of the local system.	*ftmodo -p= -l=*
TRAP	The TRAP settings are output here. The possible values are ON and OFF. The row CONS indicates the console traps. The columns designate the events for which traps may be generated: SS-STATE Change of the status of the *open*FT subsystem FT-STATE Change of the status of the *open*FT control process PART-STATE Change of the status of partner systems PART-UNREA Partner systems unreachable RQ-STATE Change of the status of request administration TRANS-SUCC Requests completed successfully TRANS-FAIL Failed requests	*ftmodo -tpc=*

Field name	Meaning and values	Command/ option
TRACE	The monitoring settings (traces) are output here. The meanings of the individual columns are as follows:	
	SWITCH Monitoring activated (ON) or deactivated (OFF)	*ftmodo -tr=*
	PARTNER-SELECTION Selection based on the partner system's protocol type. Possible protocol types: OPENFT, FTP, FTAM. ALL means that nothing is selected, i.e. all partner systems	*ftmodo -trp=*
	REQUEST-SELECTION Selection based on the request type The following are possible: ONLY-SYNC/ONLY-ASYNC (only synchronous or only asynchronous requests) ONLY-LOCAL/ONLY-REMOTE (only locally or only remotely submitted requests). ALL means that nothing is selected, i.e. all requests.	*ftmodo -trr=*
	OPTIONS: NONE: No options NO-BULK-DATA: Minimum trace, i.e. bulk data (file contents) is not logged. In addition, no repetitions of data log elements are logged.	*ftmodo -tro=*

5.29 ftshwp - Display FT profiles

tshwp stands for "show profile" and allows you to obtain information about FT profiles. In short form, it displays the names of the selected FT profiles, as well as the following information:

– whether or not the FT profile is privileged asterisk (*) before the profile name
– whether or not the transfer admission is disabled exclamation mark (!) before the profile name.

As the FTAC administrator, you may obtain information about all FT profiles in the system.

Format

ftshwp -h |
[<profile name 1..8>]
[-s=[<transfer admission 8..32> | @a | @n][,<user ID 1..32> | @a]]
[-l] [-csv]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name
Is the name of the FT profile you wish to see.

profile name not specified
Profile name is not used as a criterion for selecting the FT profile to be displayed. If you do not specify the profile with -s (see below), FTAC will display information on all of your FT profiles.

-s=[transfer admission | @a | @n][,user ID | @a]
Is used to specify criteria for selecting the FT profiles to be displayed.

Transfer admission
Is the transfer admission of the FT profile to be displayed. A binary transfer admission must be specified in hexadecimal format in the form x\'...\' or X\'...\'.

@a for *transfer admission*
Displays information either on the FT profile specified with *profile name* (see above) or (if no *profile name* was specified) on all FT profiles.

As the FTAC administrator, you can specify *@a* if you want to obtain information on FT profiles belonging to other login names, since even you should not know the transfer admission.

@n for *transfer admission*
As the FTAC administrator, you can specify *@n* if you want to obtain information on FT profiles belonging to other login names which do not have defined transfer admissions.

transfer admission not specified
causes FTAC to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time. If you just press ↵, this has the same effect as specifying *@a*.

,user ID
As the FTAC administrator, you can specify any login name here.

@a for *user ID*
As the FTAC administrator, you can obtain information on the FT profiles of all login names.

user ID not specified
displays only profiles belonging to the user's own login name, regardless of who issues the command.

-s not specified
if no profile name is specified, displays all the FT profiles belonging to the login name under which the *ftshwp* command is issued. Otherwise, displays information on the FT profile with the specified name.

-l displays the contents of the selected FT profiles.

In long form, the entire contents of the selected FT profiles are displayed. The USER-ADM parameter contains the following information:
- the login name for which it is valid
- whether or not it is valid for a specific password of the login name
- whether or not it is valid for any password of the login name
- whether or not it has an undefined password and is thus disabled.

USER-ADM=	**Meaning**
(user ID,,OWN)	Profile is valid for all passwords of the login name.
(user ID,,YES)	The profile is valid only for a specific password of the login name (specified in *-ua=user ID, password* with an *ftcrep* or *ftmodp* command). The profile is deactivated (not disabled) if the password is changed. You can activate it again, for example, by resetting the password.
(user ID,, NOT-SPECIFIED)	The FTAC administrator created or modified the FT profile knowing only the login name. As a result, the profile was disabled. You must enable the profile with *ftmodp* and the *-v=y* parameter.

If an FT profile is disabled, the *TRANS-ADM* parameter indicates the reasons why the profile was disabled. The following table shows the possible parameter values, as well as their meanings:

TRANS-ADM=	**Possible cause and action**
NOT-SPECIFIED	The FTAC administrator created the FT profile without transfer admission, or the FTAC user did not specify transfer admission. Measure: specify transfer admission
DUPLICATED	An attempt was made to create an FT profile with the same transfer admission. Measure: specify new transfer admission
LOCKED (by_adm)	The FTAC administrator modified the FT profile by login name only. The transfer admission remained unchanged but was disabled. Measure: enable the profile using the *ftmodp* command and the *-v=y* parameter
LOCKED (by_import)	The FT profile was created using the *ftimpe* command. The transfer admission remains unchanged, but is marked as disabled. Measure: enable the profile using the *ftmodp* command and the *-v=y* parameter.

TRANS-ADM=	Possible cause and action
LOCKED (by_user)	The FTAC user disabled his/her own FT profile. Measure: enable profile using the *ftmodp* command and the *-v=y* parameter.
EXPIRED	The time up to which the transfer admission can be used has expired. Measure: enable profile using the *ftmodp* command and the *-d* parameter, by removing the temporal restriction using the *-d* entry and defining a new time span with *-d=date*.

ftshwp does not, however, provide a means of displaying a transfer admission. If you have forgotten a transfer admission, you have to define a new one using *ftmodp*.

-l not specified
: displays only the names of your FT profiles. Markings also indicate whether or not an FT profile is privileged (*) and whether or not it is disabled (!).

-csv You can use *-csv* to specify that the FT profiles are to be output in the CSV format. The values in the output are separated by semicolons. If *-csv* is specified, output is always in long form (analogous to *-l*) regardless of whether or not *-l* has also been specified.

-csv not specified
: The FT profiles are output in the standard format.

5.30 ftshwptn - Display partner properties

You use the *ftshwptn* command to call up the following information about the partner systems entered in the partner list:

– The name of the partner system

– The status of the partner system (activated, deactivated)

– The security level that was assigned to the partner level

– The setting for the *open*FT monitoring function (trace) for the partner system

– The number of file transfer requests to the partner system issued in the local system that have not yet been completed

– The number of file transfer requests for the local system that have been issued in the partner system

– The mode for sender verification and authentication

– The partner system's transport address, possibly with the port number if this is different from the default

– The identification of the partner system

– The routing information if the partner system can only be accessed via an intermediate instance

You can also output the partners in the partner list as a platform-specific command sequence. In this way, it is possible to save the partner list and load it at another computer which may possibly be running a different operating system.

Format

ftshwptn -h |
[<partner 1..200>]
[-st=a | -st=na | -st=d | -st=ie | -st=nc | -st= ad | -st=da]
[-l | -csv | -px | -pw | -p2 | -pz]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner
: Specifies the partner whose properties you want to display. You can specify the name of the partner in the partner list or the address of the partner system.

partner not specified
: The properties of all the partners in the partner list are displayed.

-st= a | na | d | ie | nc | ad | da
: This operand enables you to display the properties of partner systems which have a specific status. You have the following possibilities:

  a (active)
  : All the partner systems with the status ACTIVE are displayed.

  na (not active)
  : All the partner systems which do **not** have the status ACTIVE are displayed.

  d (deactive)
  : All the partner systems with the status DEACTIVE are displayed.

  ie (installation error)
  : All the partner systems with the status LUNK, RUNK, LAUTH, RAUTH, NOKEY or IDREJ are displayed.

  nc (not connected)
  : All the partner systems with the status NOCON or DIERR are displayed.

  ad (active + automatic deactivation)
  : All the partner systems for which the option AUTOMATIC-DEACTIVATION is set (see the option *-ad* in the *ftaddptn* and *ftmodptn* commands) but are still active are displayed.

  da (deactive + automatic deactivation)
  : All the partner systems which have actually been deactivated because of the AUTOMATIC-DEACTIVATION option are displayed.

-st not specified
: The output is not restricted to partner systems with a specific status.

-l | **-csv** | **-px** | **-pw** | **-p2** | **-pz**

These options determine the scope and format of the output.

-l The properties of the partner systems are output in full as a table.

-csv The properties of the partner systems are output in CSV format. The individual values are separated by semicolons.

-px The properties of the partner systems are output as a command sequence. This can be called in UNIX systems as a shell procedure in order to generate partner entries with identical properties.

-pw The properties of the partner systems are output as a command sequence. This can be called in Windows systems as a batch procedure in order to generate partner entries with identical properties.

-p2 The properties of the partner systems are output as a command sequence. This can be called in BS2000 systems as an SDF procedure in order to generate partner entries with identical properties.

-pz The properties of the partner systems are output as a command sequence. This can be called in z/OS systems as a CLIST procedure in order to generate partner entries with identical properties.

-l, *-csv*, *-px*, *-pw*, *-p2*, *-pz* not specified

If you do not specify any of these options then the partners' properties are output in their abbreviated form.

Output format of ftshwptn

Examples

```
$ftshwptn
NAME      STATE SECLEV   TRACE  LOC   REM P-CHK ADDRESS
Testsys   ACT   STD      FTOPT  0     0 FTOPT D123S456.mydomain.com
tam01     ACT       5    FTOPT  0     0       ftam://%ip123.11.22.33
ftamfsc   ACT   STD      FTOPT  0     0       ftam://PC01.test.net:.sni-ftam
ftamdex   ACT   STD      FTOPT  0     0       ftam://PC02:102.TS1.SS1-PS1
INAKT     DEACT STD      FTOPT  0     0       INAKT
FtFtif    ACT   STD      FTOPT  0     0       UX000001:.tobs2
ftp001    ACT   STD      FTOPT  0     0       ftp://UX000002


ftshwptn -l
NAME      STATE SECLEV   TRACE  LOC   REM P-CHK ADDRESS
                                             ROUTING  IDENTIFICATION
pingftam ACT       50   OFF    0     0       ftam://PINGPONG.mynet.de:.s
                                             ni-ftam

PINGO     ACT   STD      ON     0     0 T-A   PINGPONG.mynet.de:1234
                                                     PINGPONG.mynet.de
rout0001 ACT   STD      FTOPT  0     0 FTOPT INCOGNITO
                                             ROUT01   INCOGNITO.id.new
servftp   ACT   B-P-ATTR ON     0     0       ftp://ftp.mynet.de
```

Explanation of output

NAME
: Name of the entry in the partner list.

STATE
: Specifies how file transfer requests issued locally to the specified partner system are processed.

  ACT File transfer requests issued locally to this partner system are processed with *ftstart*.

  DEACT
  : File transfer requests issued locally to this partner system are initially not processed, but are only placed in the request queue.

  ADEAC
  : Failed attempts at establishing a connection lead to this partner system being deactivated. The maximum number of failed attempts is 5. In order to perform file transfers with this partner system again, it must be explicitly reactivated with *ftmodptn -st=a*.

  NOCON
  : Attempt to establish a transport connection failed.

LUNK
: Local system is not known in the remote FT system.

RUNK
: Partner system is not known in the local transport system.

AINAC
: Partner system has been deactivated after a number of unsuccessful attempts to establish a connection.

LAUTH
: Local system could not be authenticated in the partner system. A valid public key for the local *open*FT instance must be made available to the partner system.

RAUTH
: Partner system could not be authenticated in the local system. A valid public key for the partner system must be stored in the folder */var/openFT/Instance/syskey*. *Instance* stands for the name of the relevant instance.

DIERR
: A data integrity error has been detected on the connection to the partner system. This can be the result of an error in the transport system or of attempts at manipulation on the data transfer path. The connection has been interrupted, but the affected request is still live (if it has the capability of being restarted).

NOKEY
: The partner does not accept unencrypted connections, but no key is available in the local system. A new key must be generated.

IDREJ
: The partner or an intermediate instance has not accepted the instance ID sent by the local system. Check whether the local instance ID matches the entry for the partner in the partner list.

SECLEV
: Security level assigned to the partner system.

1..100
: Security level. 1 is the lowest security level (partner is extremely trusted) and 100 is the highest security level (partner is not trusted).

STD The global setting for the security level applies.

B-P-ATTR
: The security level is assigned to the partner on the basis of the partner's attributes, i.e.:
  – Security level 10 if the partner has been authenticated.
  – Security level 90 if the partner is known in the transport system.
  – Security level 100 otherwise, i.e. if the partner has only been identified by its address.

TRACE
: The global settings for partner selection in the *open*FT trace function apply.

  FTOPT
  : The global setting for partner selection in the *open*FT trace function applies.

  ON The trace function is activated for this partner. However, a trace is only written if the global *open*FT trace function is also activated.

  OFF The trace function is deactivated for this partner.

LOC Shows the number of file transfer requests addressed to the FT system entered in the local system.

REM Shows the number of file transfer requests issued by the remote FT system and addressed to the local FT system.

P-CHK
: Shows the settings for sender verification and authentication.

  FTOPT
  : The global setting for sender verification applies.

  STD Checking of the transport address is deactivated. Only the identification of the partner is checked. The transport address of the partner is not checked even if extended sender verification is activated globally.

  T-A Checking of the transport address is activated. The transport address of the partner is checked even if checking of the transport address is deactivated globally. If the transport address used by the partner to log in does not correspond to the entry in the partner list, the request is rejected.

  AUTHM
  : Authentication is activated.

NOKEY
: No valid key is available from the partner system although authentication is required.

ADDRESS
: Address of the partner system.

ROUTING
: Routing info of the partner system if specified. The routing info is only output with *fthswptn* -l.

IDENTIFICATION
: Identification of the partner system if specified. The identification is only output with *fthswptn -l*.

5.31 ftstart - Start asynchronous *open*FT server

The asynchronous *open*FT server is started. This processes all the requests stored in the request queue as well as all the inbound requests.

When the asynchronous *open*FT server is started, the protection bit settings for files that are created on inbound requests are set implicitly. The settings for the shell under which you entered *ftstart* apply. For more details, see section “Setting the protection bit for newly created files” on page 22.

Format

ftstart [-h]

Description

-h Displays the command syntax on the screen.

5.32 ftstop - Stop asynchronous *open*FT server

This command shuts down the asynchronous *open*FT server. After this, no further inbound requests and no locally submitted asynchronous requests are processed:

– Inbound requests are rejected

– Locally submitted asynchronous requests are stored in the request queue

Once the *ftstop* command has been issued, the asynchronous *open*FT server server is not stopped until all the server processes have been terminated. This may take a few minutes if, for example, disconnection is delayed due to line problems.

When the asynchronous *open*FT server is restarted, the requests present in the request queue are processed normally. Requests that were cancelled due to the shutdown of the asynchronous *open*FT server are relaunched provided that the partner supports this function.

Format

ftstop [-h]

Description

-h Displays the command syntax on the screen.

5.33 ftupdi - Update the instance directory

Using *ftupdi*, you can update an instance file tree that was made using *open*FT < V10.0 so that it can continue to be used with *open*FT V10.0. The settings of the operational parameters, FTAC admission sets, FTAC admissions profiles and log records are retained.

Any interrupted requests for this instance which are still present will be lost.

Format

ftupdi -h |
<directory 1..128>

Description

-h Displays the command syntax on the screen. Any entries after *-h* are ignored.

directory
Here, you enter the directory which contains the instance file tree of the instance to be updated.

Messages of the ftupdi command

If *ftupdi* could not be carried out as specified, an explanatory message is displayed; the exit code will then be “not equal to zero“.

Example

The FT administrator wants to update the directory of the instance *hugo*.

```
ftupdi /var/openFT/.hugo
```

5.34 ftupdk - Update public keys

Using *ftupdk*, you can update the public key files of existing key pair sets.

For example, you can use it to insert updated comments from the *syspkf.comment* file into existing public key files or replace accidentally deleted public key files of a key pair set.

Format

ftupdk [-h]

Description

-h Displays the command syntax on the screen.

Example

The name of the FT administrator is to be imported into the public key files. First, the file *syspkf.comment* is edited using an editor. This file is located in the *config* subdirectory of the instance directory, see the *ftcrei* command on page 80.

The file might, for example, contain only the following line:

```
FT administrator: John Smith, Tel. 12345
```

The command is:

```
ftupdk
```

The command is executed without an error message. Following this, the information will be placed at the beginning of all *syspkf...* public key files as a comment line.

5.35 install.ftam - Install *open*FT-FTAM

The *install.ftam* command allows you to install and uninstall *open*FT-FTAM. Installation is only permitted if you have an *open*FT-FTAM license.

The *install.ftam* command is located in the */opt/openFT/bin/ftbin* directory.

Format

install.ftam –h | -i | -d

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i *open*FT-FTAM is installed.

-d *open*FT-FTAM is uninstalled.

5.36 install.ftp - Install *open*FT-FTP

You use the *install.ftp* command to install and uninstall *open*FT-FTP. Installation is only permitted if you have an *open*FT-FTP license.

The *install.ftp* command is located in the */opt/openFT/bin/ftbin* directory.

Format

install.ftp –h | -i | -d

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i *open*FT-FTP is installed.

-d *open*FT-FTP is uninstalled.

6 What if ...

... the BS2000 system cannot be accessed

Depending on which partner has the initiative, you should check the following points.

UNIX system to BS2000:

If your local system in BS2000 is unknown, enter the command *add-ft-partner* in BS2000.

If you receive the message “Remote system not available”, check whether one of the following reasons is the cause:

– Resource bottleneck in the remote system
– Remote FT system is not started
– BCIN is missing
– no network connection (for a TCP/IP connection, check the connection with the command *ping*, for example)
– Name server entry

BS2000 to UNIX system:

If necessary, check whether one of the following reasons is the cause in the BS2000 system:

– BCMAP entry is missing or invalid
– BCIN is missing
– Partner entry (*add-ft-partner*) refers to a wrong BCAM name
– BCACT is missing. Test the availability of a partner with a synchronous command (e.g *SHOW-REM-FILE-ATTRIBUTE*)

... the name of the partner is missing in the log records

Enter the partner in the partner list, in the TNS, in the DNS or */etc/hosts*.

... the logging function cannot be called or the log file is corrupted

The only remedy here, is to terminate *open*FT (*ftstop*) and delete the log file as follows:

```
rm /var/openFT/instance/log/logdat
```

However, this means that you lose all log records. Here *instance* means the name of the corresponding instance.

... access to the admission set and admission profile file causes errors or if this file is defective

The possible reasons are:

1. Manual access to *sysfsa.dat* and/or *sysfsa.idx* (both files are located in the instance directory under *config*).
2. System crash or *kill* of *open*FT process with *sysfsa.** open
3. File system full on ISAM access

In cases 2 and 3, ISAM reacts negatively and usually leaves an unusable index file.

Possible solutions:

– Attempt to export/import:
  Use *ftexpe* to export the data to a backup file.
  Then shut down the *open*FT server with *ftstop*, delete *sysfts.da*t and *sysfsa.idx* and restart *open*FT with *ftstart*. Import *ftimpe* from the backup file.

– Try to repair the ISAM index file with *dcheck* (*inst-dir* is the instance directory):

  ```
  /opt/openFT/bin/ftbin/dcheck -b inst-dir/config/sysfsa
  ```

  It may be necessary to delete the index file explicitly:

  – If the data file *sysfsa.dat* is empty then no data is lost. As a result, both ISAM files can be deleted with *open*FT stopped and can then be initialized before *ftstart* by using the *ftshwa* command.

  – If the data file already contains modifications to the admission sets and/or profiles then you should enter the following commands:

  ```
  cd /var/openFT/<instance>/config
  ftstop
  mv sysfsa.dat sav.sysfsa.dat && rm sysfsa.idx
  ftshwa >/dev/null
  rm sysfsa.dat && mv sav.sysfsa.dat sysfsa.dat
  /opt/openFT/bin/ftbin/dcheck -b sysfsa
  ```

```
ftstart
```

Explanation:
If *sysfsa.idx* is defective, it must be recreated. To do this, you must first back up the *sysfsa.dat* file that you want to create. You then use *ftshwa* to create a new *sysfsa.dat* file which you immediately delete and replace with the backed up *sysfsa.dat* file. The resulting file pair can now be re-used.

– If this attempt also fails, you must delete the admission set and admission profile and make new entries to ensure a consistent state.

... You are not given a free transport connection for an ncopy request

– Check the partner address in the partner entry or in the partner list

– If you are working with TNS: check your TNS entries and check whether TNS use is activated (in the case of *ftshwo*, the value *YES must be displayed for USE TNS; otherwise activate TNS use with *ftmodo -tns=y*).

– Check the address settings in the operating parameters

... the *open*FT message “Remote transfer admission invalid” appears

For reasons of data security, this message does not differentiate between the various possible reasons for the rejection on the initiator side. This information is only available via the *open*FT logging of the responder system.

... Do requests still remain in the “WAIT” state?

– Check whether the asynchronous *open*FT server is started in the local system

– Check whether the *open*FT or asynchronous *open*FT server is started in the remote system

Using *ftshwr -l,* you can obtain further information on the cause.

.. Deleting a request in the *open*FT Explorer takes an unusually long time (about 1 minute)

This may mean

– that a request was issued to send a mail when the request to be deleted is finished

– and that the mail function of the UNIX system takes about 1 minute to send a mail due to a configuration problem.

Solution:

Do not ask for a mail to be sent when the request is finished, i.e. specify the *-m=n* option for the *ft* command. Requests that are started in the *open*FT Explorer never require a mail to be sent when finished.

... in Linux systems, the left mouse button does not function as desired in the Explorer

This may be due to the fact that the function of the NumLock key was set differently on generation in Linux with Xfree and KDE (in larger SuSE Linux systems).

This causes problems if the NumLock key functions as an Alt Lock key: a click then becomes an Alt-click and a double-click becomes an Alt-double-click.

The administrator can overcome this problem by toggling the NumLock key. It may also be possible to set the Numlock functionality in the BIOS.

Performance note

The RFC1006 protocol is far more efficient than communicating via LANINET. If you use the TNS (*ftmodo -tns=y*), you should therefore set the RFC1006 protocol for TNS entries in UNIX systems. In BS2000, the type of the global BCMAP entry determines the protocol type: if the PTSEL-I entry exists, RFC1006 is used.

6.1 Actions in the event of an error

If, in spite of precautions, an error occurs which neither the FTAC administrator nor the system administrator can rectify, please contact your local Fujitsu Siemens Computers contact partner. In order to simplify error diagnosis, you should provide the following documents:

– an exact description of the error situation and information as to whether the error is reproducible;

– the version number of the file transfer product in the remote computer;

– diagnostic information (which is created with the FT command *ftshwd*);

– if available, the FTAC and FT log records (which are output with the FT command *ftshwl ...*);

– if available, the *open*FT trace file;

– for errors related to a specific FT profile a printout of the profile (*ftshwp␣profilename␣-l*) and a printout of the admission sets (*ftshwa␣@a*).

– version of the operating system

– version of the communication system (CMX, PCMX etc.)

– if necessary, the process tables (*ps* command)

You can also call the procedure */opt/openFT/bin/ftbin/ftdiaginfo* to initiate the collection of various diagnostic data. You should then send the resulting tar file together with a description of the error to the responsible contact person.

7 Diagnosis

This chapter describes how you can create and evaluate trace files.

Further diagnostic information can be obtained with the help of the command “ftshwd - Display diagnostic information” on page 134.

At the end of this chapter you will find code tables with which you can diagnose code conversion errors.

7.1 Trace files

You can switch trace mode on or off for the purposes of error diagnosis.

7.1.1 Activating/deactivating trace functions

You can activate and deactivate the trace function as follows:
– the *ftmodo -tr=n/f* command
– or the graphical interface.

When trace mode is switched on, diagnostic data is written to trace files which are located in the directory */var/openFT/std/traces* or, if the traces were created by another *open*FT instance, in the subdirectory *traces* residing in the corresponding instance directory. When you have finished diagnosis, you should deactivate the trace mode for reasons of performance. The trace files can become infinitely large, since they are not cyclically overwritten. However, you can also close trace files with the *ftmodo -tr=c* command and open new trace files.

7.1.2 Viewing trace files

You can either view trace files directly in the graphical user interface or open them in an editor after preparing them with the *fttrace* command.

Files which have the suffix *.fttf* are protocol trace files. They begin with *Y*, *S*, *N* or *C*. Files with the suffix *.PPE* are interface trace files.

The names of the trace files have the following format:

– Y*oddhhmm*.S*ssccc*.P*ppppp*.fttf
Protocol trace files for synchronous outbound requests.

– S*oddhhmm*.S*ssccc*.P*ppppp*.fttf
  Protocol trace files for asynchronous outbound requests and inbound requests.

– N*oddhhmm*.S*ssccc*.P*ppppp*.fttf
  Protocol trace files for commands.

– C*oddhhmm*.S*ssccc*.P*ppppp*.fttf
  Protocol trace files for the control process.

– *process*-*pid*-*thid*-*time*.PPE
  Interface trace files. Here, *process* is the name of the process which the command has executed, *pid* the process ID as a hexadecimal number, *thid* the thread ID as a hexadecimal number and *time* the time in milliseconds since the system start.

The specification *oddhhmm*.S*ssccc*.P*ppppp* in the protocol trace file names indicates the creation time and the process ID. Here, *o* indicates the month (1 = January, 2 = February, ... A= October, B=November, C = December), *dd* the day, *hhmm* the time in hours (hh) and minutes (mm), *ssccc* the time in seconds (ss) and milliseconds (ccc) and *ppppp* the process ID.

7.1.3 Evaluating trace files with fttrace

Trace files for all protocols (*open*FT, FTAM and ftp protocol) are evaluated with the *fttrace* command.

Format

```
fttrace -h |
        [ -d ]
        [ -sl=n | -sl=l | -sl=m | -sl=h ]
        [ -cxid=<context id> ]
        [ -f=hh:mm:ss ]
        [ -t=hh:mm:ss ]
          <trace files>
```

Description

-h Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-d Specifies that the trace files are to be output in hexadecimal format (dump format).

If you do not specify *-d* then the files are output in printable form, default value.

-**sl**=n | -**sl**=l | -**sl**=m | -**sl**=h
Specifies the security level for the output..

n (no) No security requirements, i.e. all the data is output. This includes IDs, passwords, file names etc.

l (low) Passwords are overwritten with XXX.

m (medium)
Passwords, user IDs, account numbers and follow-up processing commands are overwritten with XXX, default value.

h (high)
Passwords, user IDs, account numbers, follow-up processing commands and file names are overwritten with XXX.

-cxid=context id
Selects the trace entries on the basis of the context ID. In this context ID, the first position is the identifier of the slot pool and the second to fourth positions identify the slot. If you omit *-cxid* or specify *-cxid*= without a context ID then trace entries are output for all context IDs.

-f=hh:mm:ss (from)
Specifies the time as of which trace entries in the trace file are to be evaluated. You enter the time in the format hours:minutes:seconds (2 digits each).

If you do not specify a start time then trace entries are output from the start of the file.

-t=hh:mm:ss (to)
Specifies the time up to which trace entries in the trace file are to be evaluated. You enter the time in the format hours:minutes:seconds (2 digits each).

If you do not specify an end time then trace entries are output up to the end of the file.

trace files
Name(s) of the trace file(s) that you want to evaluate. You can specify multiple trace files and wildcards can be used.

7.2 Code tables

7.2.1 Code table EBCDIC.DF.04

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0					SP	&	-	¢	Ø	°	µ	¢	ù	¦	Ù	0
1					NBSP	é	/	É	a	j	‾	£	A	J	³	1
2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
3					ä	ë	Ä	Ë	c	l	t	•	C	L	T	3
4					à	è	À	È	d	m	u	©	D	M	U	4
5					á	í	Á	Í	e	n	v	§	E	N	V	5
6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
7					å	ï	Å	Ï	g	p	x	1/4	G	P	X	7
8					ç	ì	Ç	Ì	h	q	y	1/2	H	Q	Y	8
9					ñ	ß	Ñ	••	i	r	z	3/4	I	R	Z	9
A					‘	!	^	:	<<	ª	;	¬	SHY	1	2	3
B					.	$	,	#	>>	º	¿	[	ô	û	Ô	{
C					<	*	%	@	ð	æ	Ð	\	ö	ü	Ö	Ü
D					(	)	_	‘	ý	,	Ý	]	ò	Û	Ò	}
E					+	;	>	=	Þ	Æ	þ	´	ó	ú	Ó	Ú
F							?	“	±	¤	®	x	õ	ÿ	Õ	~

Code table EBCDIC.DF.04 (character set corresponding to ISO 8859-1)

7.2.2 Code table ISO 8859-1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	`	p			NBSP	°	À	Ð	à	ð
1			!	1	A	Q	a	q			¡	±	Á	Ñ	á	ñ
2			"	2	B	R	b	r			¢	²	Â	Ò	â	ò
3			#	3	C	S	c	s			£	³	Ã	Ó	ã	ó
4			$	4	D	T	d	t			¤	´	Ä	Ô	ä	ô
5			%	5	E	U	e	u			¥	m	Å	Õ	å	õ
6			&	6	F	V	f	v			¦	¶	Æ	Ö	æ	ö
7			´	7	G	W	g	w			§	•	Ç	x	ç	³
8			(	8	H	X	h	x			"	,	È	Ø	è	ø
9			)	9	I	Y	i	y			©	¹	É	Ù	é	ù
A			*	:	J	Z	j	z			ª	º	Ë	Ú	ê	ú
B			+	;	K	[	k	{			«	»	Ê	Û	ë	û
C			,	<	L	\	l	|			¬	1/4	Ì	Ü	ì	ü
D			-	=	M	]	m	}			SHY	1/2	Í	Ý	í	ý
E			.	>	N	^	n	~			®	3/4	Î	Þ	î	þ
F			/	?	O	-	o				¯	¿	Ï	b	ï	ÿ

Code table ISO 8859-1

8 Appendix

8.1 Structure of CSV Outputs

8.1.1 ftshwa

The following table indicates the CSV output format of an admission set.

Column	Type	Values
UserId	String	Value enclosed in double quotes
UserMaxObs	Number	Value
UserMaxObsStd	String	*YES / *NO
UserMaxObr	Number	Value
UserMaxObrStd	String	*YES / *NO
UserMaxIbs	Number	Value
UserMaxIbsStd	String	*YES / *NO
UserMaxIbr	Number	Value
UserMaxIbrStd	String	*YES / *NO
UserMaxIbp	Number	Value
UserMaxIbpStd	String	*YES / *NO
UserMaxIbf	Number	Value
UserMaxIbfStd	String	*YES / *NO
AdmMaxObs	Number	Value
AdmMaxObsStd	String	*YES / *NO
AdmMaxObr	Number	Value
AdmMaxObrStd	String	*YES / *NO
AdmMaxIbs	Number	Value
AdmMaxIbsStd	String	*YES / *NO
AdmMaxIbr	Number	Value
AdmMaxIbrStd	String	*YES / *NO
AdmMaxIbp	Number	Value

Column	Type	Values
AdmMaxIbpStd	String	*YES / *NO
AdmMaxIbf	Number	Value
AdmMaxIbfStd	String	*YES / *NO
Priv	String	*YES / *NO
Password	String	*YES / *NO

8.1.2 ftshwl

The following table indicates the CSV output format of a logging record.

A format template in Microsoft Excel format is present in the file */opt/openFT/samples/ftaccnt.xlt* as an example of a possible evaluation procedure.

Column	Type	Value
LogId	Number	Value
ReasonCode	String	Value enclosed in double quotes to prevent interpretation as a number. FTAC Reason Codes are output as Hex strings as in the manual (in contrast to OPS variables)
LogTime	yyyy-mm-dd hh:mm:ss	Value
InitUserId	String	Value enclosed in double quotes / *REM
InitTsn	String	Value enclosed in double quotes / *NONE
PartnerName	String	Value enclosed in double quotes
TransDir	String	*TO / *FROM / *NSPEC
RecType	String	*FT / *FTAC

Column	Type	Value
Func	String	*TRANS-FILE / *READ-FILE-ATTR / *DEL-FILE / *CRE-FILE / *MOD-FILE-ATTR / *READ-DIR / *MOVE-FILE / *CRE-FILE-DIR / *DEL-FILE-DIR / *MOD-FILE-DIR
UserAdmisId	String	Value enclosed in double quotes
FileName	String	Value enclosed in double quotes
Priv	String	*NO / *YES for FTAC log records and entry of an FTAC profile; otherwise *NONE
ProfName	String	Value enclosed in double quotes / *NONE
ResultProcess	String	*NONE / *STARTED / *NOT-STARTED
StartTime	yyyy-mm-dd hh:mm:ss	Value
TransId	Number	Value
Write	String	*REPL / *EXT / *NEW / *NONE
StoreTime	yyyy-mm-dd hh:mm:ss	Value
ByteNum	Number	Value
DiagInf	String	Value enclosed in double quotes / *NONE
ErrInfo	String	Value enclosed in double quotes / *NONE
SecEncr	String	*YES or *NO
SecDichk	String	*YES or *NO
SecDencr	String	*YES or *NO
SecDdichk	String	*YES or *NO
SecLauth	String	*YES or *NO

Column	Type	Value
SecRauth	String	*YES or *NO
RsaKeyLen	Number	Value, the space remains empty if SecEncr does not have the value *YES
SymEncrAlg	String	DES or AES, the space remains empty if SecEncr does not have the value *YES
CcsName	String	Value

8.1.3 ftswho

The following table indicates the CSV output format of the operating parameters..

Column	Type	Value
PartnerLimit	Number	Value
ReqLim	Number	Value
TaskLim	Number	Value
ConnLim	Number	Value
ReqWaitLev	Number	Value
TransportUnitSize	Number	Value
PartnerCheck	String	*STD / *TRANSP-ADDR
SecLev	Number	*B-P-ATTR / Value
TraceOpenft	String	*STD / *OFF
TraceOut	String	*FILE / *OFF
TraceSession	String	*STD / *OFF
TraceFtam	String	*STD / *OFF
LogTransFile	String	*ON / *OFF
MaxInboundReq	Number	Value
MaxReqLifetime	String	Value / *UNLIMITED
SnmpTrapsSubsystemState	String	Empty[1]
SnmpTrapsFtState	String	Empty[1]
SnmpTrapsPartnerState	String	Empty[1]
SnmpTrapsPartnerUnreach	String	Empty[1]
SnmpTrapsReqQueueState	String	Empty[1]
SnmpTrapsTransSucc	String	Empty[1]
SnmpTrapsTransFail	String	Empty[1]
ConsoleTraps	String	*ON / *OFF
TeleService	String	Empty[1]
HostName	String	Value / *NONE
Identification	String	Value enclosed in double quotes

Column	Type	Value
UseTns	String	*YES / *NO
ConsTrapsSubsystemState	String	*ON / *OFF
ConsTrapsFtState	String	*ON / *OFF
ConsTrapsPartnerState	String	*ON / *OFF
ConsTrapsPartnerUnreach	String	*ON / *OFF
ConsTrapsReqQueueState	String	*ON / *OFF
ConsTrapsTransSucc	String	*ON / *OFF
ConsTrapsTransFail	String	*ON / *OFF
FtLog	String	*ALL / *FAIL / *NONE
FtacLog	String	*ALL / *FAIL / *NONE
Trace	String	*ON / *OFF
TraceSelp	String	*ALL / OPENFT / FTP / FTAM / *NONE
TraceSelr	String	*ALL / ONLY-SYNC / ONLY-ASYNC / ONLY-LOCAL / ONLY-REMOTE
TraceOpt	String	*NONE / *BULK-DATA
KeyLen	Number	Value
CcsName	String	Value enclosed in double quotes
AppEntTitle	String	*YES / *NO
StatName	String	Value
SysName	String	Value
FtStarted	String	*YES / *NO
openftAppl	String	Value / *STD
ftamAppl	String	Value / *STD
FtpPort	Number	Value / empty
ftstdPort	String	Value / *STD
DynPartner	String	*ON / *OFF

[1]not relevant for UNIX systems.

8.1.4 ftshwp

The following table indicates the CSV output format of an admission profile.

Column	Type	Value
ProfName	String	Value enclosed in double quotes
Priv	String	*YES / *NO
TransAdm	String	*NSPEC / *SECRET
Duplicated	String	*YES / *NO
LockedByImport	String	*YES / *NO
LockedByAdm	String	*YES / *NO
LockedByUser	String	*YES / *NO
Expired	String	*YES / *NO
ExpDate	yyyy-mm-dd	Value / *NRES
Usage	String	*PUBLIC / *PRIVATE / *NSPEC
IgnObs	String	*YES / *NO
IgnObr	String	*YES / *NO
IgnIbs	String	*YES / *NO
IgnIbr	String	*YES / *NO
IgnIbp	String	*YES / *NO
IgnIbf	String	*YES / *NO
Initiator	String	*LOC / *REM / *NRES
TransDir	String	*FROM / *TO / *NRES
MaxPartLev	Number	Value / *NRES
Partners	String	One or more FT partners, delimited by commas and enclosed in double quotes / *NRES
FileName	String	Value enclosed in double quotes / *NRES
Library	String	*YES / *NO / *NRES / Value enclosed in double quotes
FileNamePrefix	String	*YES / *NO
ElemName	String	Value enclosed in double quotes / *NRES / *NONE

Column	Type	Value
ElemPrefix	String	*YES / *NO
ElemVersion	String	Value enclosed in double quotes / *STD / *NONE / *NRES
ElemType	String	Value enclosed in double quotes / *NRES / *NONE
FilePass	String	*YES / *NRES / *NONE
Write	String	*NEW / *EXT / *REPL / *NRES
UserAdmId	String	Value enclosed in double quotes
UserAdmAcc	String	Value enclosed in double quotes / *NSPEC / *NRES
UserAdmPass	String	*OWN / *NSPEC / *NONE / *YES
ProcAdmId	String	Value enclosed in double quotes / *NRES / *SAME
ProcAdmAcc	String	Value enclosed in double quotes / *NRES / *SAME
ProcAdmPass	String	*NONE / *YES / *NRES /*SAME
SuccProc	String	Value enclosed in double quotes / *NONE / *NRES / *EXPANSION
SuccPrefix	String	Value enclosed in double quotes / *NONE
SuccSuffix	String	Value enclosed in double quotes / *NONE
FailProc	String	Value enclosed in double quotes / *NONE / *NRES / *EXPANSION
FailPrefix	String	Value enclosed in double quotes / *NONE
FailSuffix	String	Value enclosed in double quotes / *NONE
TransFile	String	*ALLOWED / *NOT-ALLOWED
ModFileAttr	String	*ALLOWED / *NOT-ALLOWED
ReadDir	String	*ALLOWED / *NOT-ALLOWED
FileProc	String	*ALLOWED / *NOT-ALLOWED
Text	String	Value enclosed in double quotes/ *NONE
DataEnc	String	*NRES / *YES / *NO

8.1.5 ftshwptn

The following table indicates the CSV output format of a partner

Name	Type	Value
PartnerName	String	Value enclosed in double quotes/
Sta	String	*ACT / *DEACT / *NOCON / *LUNK / *RUNK / *ADEAC / *AINACT / *LAUTH / *RAUTH / *NOKEY / *DIERR / *IDREJ
SecLev	String	*STD / *B-P-ATTR / Value enclosed in double quotes
Trace	String	*FTOPT / *STD / *ON / *OFF
Loc	Zahl	Value
Rem	Zahl	Value
Processor	String	Value enclosed in double quotes/ empty
Entity	String	Value enclosed in double quotes/ empty
NetworkAddr	String	Value enclosed in double quotes/
Port	Integer	Value
PartnerCheck	String	*FTOPT / *STD / *TRANSP-ADDR / *AUTH / *AUTHM
TransportSel	String	Value enclosed in double quotes
LastAccessDate	yyyy-mm-dd	Value
SessionSel	String	Value enclosed in double quotes/ empty
PresentationSel	String	Value enclosed in double quotes/ empty
Identification	String	Value eingeschlossen in doppelte Hochkommas
SessRout	String	Value enclosed in double quotes/ *ID /empty
PartnerAddr	String	Value enclosed in double quotes
Check	String	*FTOPT / *STD / *TRANSP-ADDR
AuthMand	String	*YES / *NO

8.2 Important CMX commands

This section contains a short description of the most important CMX commands needed for the *open*FT configuration. You will find detailed information in the manual „CMX Operation and Administration“.

tnsxcom - Create the TS directory

With the *tnsxcom* command you can transfer files in the *tnsxfrm* format to TS directories. You can set different modes for functions such as the syntax check, update or recreating the TS directory.

The command has the following syntax (abbreviated):

tnsxcom [**-l -s -S -u -i**] [file]

The options have the following meanings:

-l LOAD mode
tnsxcom takes the entries our of the file *file* one at a time and fills the (previously empty) TS directory with the syntactically correct entries.

-s CHECK mode
tnsxcom only applies the syntax check to the file *file* and records any possible syntax errors. The TS directory is not changed.

-S CHECK-UPD mode
Like for the *-s* option, the syntax check is run on the entire file *file* in the first run. If no syntax errors are found, then *tnsxcom* updates the TS directory in a second run.

-u UPDATE mode
tnsxcom takes the entries our of the file *file* one at a time and merges the syntactically correct entries in the TS directory. Missing entries are created and existing entries are updated during this process.

-i INTERAKTIVE mode
tnsxcom reads entries in the *tnsxfrm* format from stdin after it has indicated it is ready to receive input by outputting a prompt and merges them in the TS directory. Missing entries are created and existing entries are updated during this process.

file The name of the file with the entries in the *tnsxfrm* format that are to be evaluated when the *-l*, *-s*, *-S* or *-u* options are specified. You can specify more than one file.

Example

The following call transfers the entries in the file *input.dir* to the current TS directory:

```
tnsxcom -S input.dir
```

tnsxprop - Output properties of TS applications

tnsxprop outputs all values of all properties that are located in a TS directory for the specified TS applications to stdout in a printable format.

You can specify in which format the properties are to be output using the first parameter.

The TS applications are determined by the parameter values for *name*. The parameter values for *name* can also be passed to *tnsxprop* from the file *file*. If no data was specified for *name* or *file*, then *tnsxprop* prepares the properties of all TS applications in the TS directory in the specified format.

The command has the following syntax (abbreviated):

tnsxprop [**-S** | **-h**] [**-f** file] [name ...]

-S This is the default setting. This option can be used to output the properties in symbolic form in the *tnsxfrm* format.

-h This option can be used to prepare the properties in hexadecimal form. The output is a string of hexadecimal digits together with the corresponding bit representation in which the lowest valued bit is located on the far right.

-f file

You specify for *file* the name of a file that contains the GLOBAL NAMES of the TS application whose properties are to be queried. The GLOBAL NAMES are to be specified as described under *name*.

name The GLOBAL NAME of the TS application in the TS directory is to be specified as follows for *name*:

NP5.NP4,NP3.NP2.NP1

The individual NP*i's* are the name attributes of the GLOBAL NAME.

NP5 is name attribute [5], i.e. it is the part of the name of the lowest hierarchy level. NP1 is name attribute [1], i.e. it is the part of the name of the highest hierarchy level. The name attributes are to be specified in ascending order hierarchically from left to right.

If one of the name attributes for a GLOBAL NAME does not contain data (e.g. NP4) and a name attribute of a higher level follows this name attribute (e.g. NP3), then only the separator (.) is to be specified for the name attribute that does not contain data. A series of separators at the end of the value of *name* does not have to be specified.

If the name attributes contain special characters whose special meaning would cause the syntax to take on multiple meanings, then these special characters must be delimited using the backslash (\). When in doubt, you should delimit every special character. Superfluous characters are ignored by *tnsxprop*.

If you specify an asterisk (*) for a name attribute, then *tnsxprop* returns the properties of all TS applications that match all other name attributes specified in *name* (TS_RESTRICTED filter mode).

Examples

1. The properties of the TS application that only has name attribute [5] set to the value *example_1* are to be output in hexadecimal form:

   ```
   tnsxprop –h example_1
   ```

2. The properties of the TS application that only has name attribute [5] set to the value *example_1* are to be output in symbolic form:

   ```
   tnsxprop example_1
   ```

3. The properties of all TS applications are to be output to a file *tns*:

   ```
   tnsxprop > tns
   ```

8.3 Entering transport system applications in the TNS

As of *open*FT V10, it is no longer necessary to use the TNS for linking over TCP/IP. If you nevertheless use the TNS; for instance if you link to transport systems other than TCP/IP or you wish to make use of existing TNS entries, you must do this by setting the operational parameters, e.g. using *ftmodo -tns=y*.

The TNS identifies a transport system application (TS application) by means of a symbolic name known as the GLOBAL NAME. The symbolic name generally consists of up to five name parts.

These symbolic names are assigned address information. The necessary specifications, such as station name, application name, port number, etc. can be obtained from your network administrator.

Depending on the installation variant, (initial, full or update installation) and the type of link, certain entries are made or modified during the installation of *open*FT; see also the section “TNS entries created automatically” on page 194.

Otherwise, you must make the entries yourself. The entries in the TNS can be made with the aid of the TNS compilers *tnsxcom*. To do this, enter the TS applications in a file, and then translate this file with the aid of the TNS compilers *tnsxcom* (see the section “tnsxcom - Create the TS directory” on page 189).

If you have installed CMX, you may also enter partner applications via a menu. Note, however, that only the CMX GUI can be used for FTAM partner applications. For further details, refer to the CMX manual.

It can also be useful to enter the remote TS applications of the partner systems which are to issue requests to the local system. In *open*FT partner version 8.1 and later, ensure that the name, by which requests are processed with this partner, correspond to the instance ID of the remote system. If there is any doubt, a TNS input is required.

In this case, In the case of WAN partners, the partner is easier to identify for requests issued in the remote system. For example, the name of the partner as entered in the TNS is recorded in the log records. With FTAM partners, an entry in the TNS is the precondition for automatic restart.

Which entries are created or modified for which installation variant and which type of link are explained in the following section entitled “TNS entries created automatically”.
The procedure for the entry of remote TS applications is explained starting on page 197.

TNS entries for cluster configurations

Please note that cluster configurations are only supported for TCP/IP. You will therefore need to check all *open*FT-specific TNS entries for cluster configurations and delete those transport system entries that are not related to TCP/IP. (i.e. everything except for RFC1006 and LANINET). You will find an example of this in the appendix.

8.3.1 TNS entries created automatically

During the installation of *open*FT, depending on the installation variant, certain FT applications are automatically entered in the TNS or the existing entries are modified.

It is generally advisable not to modify the applications entered during the installation. If this is required in any case, it must be ensured that the port number of the $FJAM entry is divisible by 100 and that the port number of the $FJAMOUT entry is equal to the port number of the $FJAM entry + 1. If your system is protected by a firewall and is to be accessible from the outside, the $FJAM input port must be released in the firewall.

Initial installation

For an initial installation, the following TNS entries are made automatically (see also the file */opt/openFT/config/tnsstd*):

```
$FJAM\
        TSEL    WANNEA T'$FJAM'
        TSEL    LANSBKA T'$FJAM'
        TSEL    WANSBKA T'$FJAM'
        TSEL    OSITYPE T'$FJAM'
        TSEL    RFC1006 T'$FJAM'
        TSEL    LANINET A'1100'

$FJAMOUT\
        TSEL    WANNEA T'$FJAMOUT'
        TSEL    LANSBKA T'$FJAMOUT'
        TSEL    WANSBKA T'$FJAMOUT'
        TSEL    OSITYPE T'$FJAMOUT'
        TSEL    RFC1006 T'$FJAMOUT'
        TSEL    LANINET A'1101'

$FTAM\
        PSEL    V''
        SSEL    V''
        TSEL    LANSBKA T'$FTAM'
        TSEL    WANSBKA T'$FTAM'
        TSEL    OSITYPE T'$FTAM'
        TSEL    RFC1006 T'$FTAM'
        TSEL    LANINET A'4800'
```

The local TS application $FJAM is the contact for inbound requests from *open*FT partners, $FJAMOUT for outbound requests to *open*FT partners.

If you want set up links via TRANSIT-LU0 (EMSNA), establish the link over *open*FTIF, see the section “Example entries for linking with openFT for z/OS over openFTIF” on page 207).

The local TS application $FTAM is the contact for all inbound and outbound requests with FTAM partners.

Full installation, update installation

With a full installation, the existing TNS entries are modified as follows:

- If an entry with the name $FJAM_OUBOUND exists, it is renamed to $FJAMOUT.
- If no entry with the name $FJAM_OUTBOUND exists, but one exists with the name $FJAM001, the latter is renamed to $FJAMOUT.
- Entries from $FJAM002 and up to $FJAM016 are deleted.
- If no entry exists with the name $FJAM_OUTBOUND or $FJAM001, an entry with the name $FJAMOUT is created.
- If no $FTAM entry exists, a standard entry is created for $FTAM. The local TS application $FTAM is the contact for all inbound and outbound requests with FTAM partners.

8.3.2 Definition of the local TS application for *open*FT-FTAM

If you wish to use *open*FT-FTAM, the local application $FTAM must be defined. This is done automatically during initial installation or full installation, and also for update installation if no $FTAM entry is present. This application is used for all request with FTAM partners (outbound and inbound).

Special points

With the TCP/IP-LAN transport system, two entries must be made for the symbolic name:

- an RFC1006 entry with the transport selector. Enter the relevant symbolic name $FTAM as transport selector. The entry must be made TRANSDATA format (indicator *T*).
- a LANINET entry with the port number. The port number is specified in ASCII format.

More details on this topic can be found in the CMX manual and in “Appendix” on page 179.

You must make the entry in a defined format (see samples).

The GLOBAL NAME $FTAM is fixed. `T'$FTAM'` is recommended for the transport selector. The entries `PSEL V''` and `SSEL V''` are absolutely necessary.

Sample entries for *open*FT-FTAM on Sparc Solaris

```
$FTAM\
 PSEL    V''                    ; empty presentation
 SSEL    V''                    ; empty session selector
 TSEL    WANSBKA T'$FTAM'       ; entry for WAN-CONS, ISDN-CONS
 TSEL    LANSBKA T'$FTAM'       ; entry for ETHN-CLNS/passive
                                ; necessary for link to CMX V3.0
 TSEL    OSITYPE T'$FTAM'       ; entry for ETHN-CLNS/active
 TSEL    RFC1006 T'$FTAM'       ; entry for TCP/IP-RFC1006
 TSEL    LANINET A‘4800‘        ; entry for TCP/IP
```

8.3.3 Definition of a remote TS application for *open*FT

In *open*FT partners with version 8.1 and later, you must ensure that the name, by which requests are processed with this partner, correspond to the instance ID of the remote system. If there is any doubt, a TNS input, whose global name is the instance ID, is needed.

For each further partner system which is to be accessible for requests issued locally, it is necessary to make a TNS entry. In both of the cases described above, additional TNS entries must be made for the partner systems, and separate names assigned to the partner systems. The entries are made in the menu system or translated using the TNS compiler *tnsxcom*.

As symbolic name (GLOBAL NAME), you must use an alphanumeric name containing up to 78 characters. No special characters may be used, except for::

- "." as separator
- "#" . The entry behind the hash "#" is used to differentiate entries with the same prefix. In this way, it is possible to enter a partner (who has several addresses) several times with the same name (prefix). This is only useful for inbound requests. Here, the partner system is always displayed with the same partner address (corresponding to the prefix).

You are free to select the symbolic name. However, it must be unique in the local system. The further entries to be made depends on the how the remote system is connected to the network. The entries must be made in TRANSDATA format (indicator *T*). You can obtain the information required to make the entries from the network administrator.

8.3.3.1 Sample entries for *open*FT partners

- Entry of a PCMX partner dress for transfer via TCP/IP-RFC1006 and a PCMX, CMX-V4.0 or Windows partner (as of FT-PCD V2.6):

```
ftrfc\
        TA      RFC1006 123.4.5.67     PORT 1100 T'$FJAM'
;                       Internet addr. Portno    T selector
```

- Entry of variable Internet addresses for one and the same partner with the name *mobile* (e.g. a Notebook used from different locations and thus connected via different Internet addresses):

```
mobile\
        TA      RFC1006 100.22.33.45     PORT 1100 T'$FJAM'
;                       Internet-addr1.  Portno   T selector
mobile#1\
        TA      RFC1006 101.20.30.40     PORT 1100 T'$FJAM'
;                       Internet addr2.  Portno   T selector
mobile#2\
        TA      RFC1006 102.21.31.41     PORT 1100 T'$FJAM'
;                       Internet-addr3.  Portno   T selector
```

- Entry of a partner address (*open*FT for BS2000/OSD partners) for transfer via TCP/IP-RFC1006 (Port 102):

```
ftbs2\
        TA      RFC1006 123.4.5.68     T'$FJAM'
;                       Internet addr. T selector
```

- Entry of a partner address for transfer via ETHN-CLNS/active:

```
ftethna\
        TA      OSITYPE 49+006C080015304050FE T'$FJAM'
 ;                      OSI network addr. T selector
```

  (OSI network address as per ISO Standard 8348/Add.2, the structure is described in the CMX manual.)

- Entry of a partner address for transfer via ETHN-CLNS/passive:

```
ftethnp\
        TA      LANSBKA 080014110960  T'$FJAM'
 ;                      Ethernet addr. T selector
```

- Entry of a partner address for transfer via WAN-NEA, WAN-NX25, ISDN-NEA, ISDN-NX25

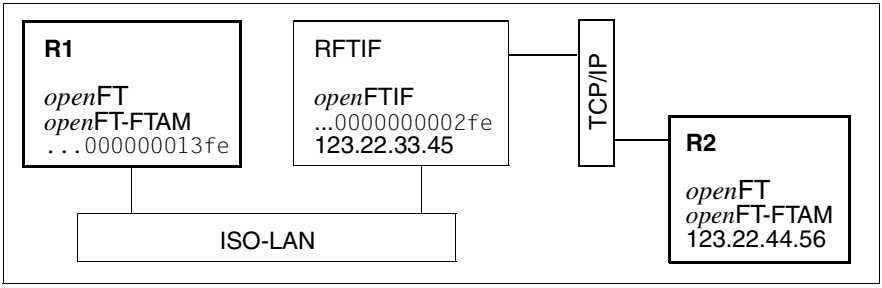
```
ftwannea\
        TA      WANNEA T'$FJAM'    1/18            WAN 2
;                     T selector  Proc./region    WAN CC
```

- Entry of a partner address for transfer via WAN-CONS, ISDN-CONS

```
    ftcons\
           TA     WANSBKA X.121 45890012233 T'$FJAM'  WAN 3
;                            SNPA info       T Sel.    WAN CC
```

8.3.3.2 *open*FTIF example for linking two UNIX systems via *open*FT protocol

In the following example, the two UNIX systems R1 and R2 are linked with the aid of a gateway computer RFTIF (with *open*FTIF for UNIX systems software) via an ISO-LAN and a TCP/IP-N network. File transfer is possible in both directions between the two processors. SMAWcmx is used on computers R1 and RFTIF and PCMX is used on the computer R2.



The section below describes all TNS entries in the processors R1, R2 and RFTIF required for file transfer between R1 and R2.

TNS entries in processor R1:

```
$FJAM\
     TSEL    OSITYPE T'$FJAM'
$FJAMOUT\
      TSEL    OSITYPE T'$FJAMOUT'
ftr2\
      SSEL   A'FTIFR2' TA OSITYPE
      470058+014445010000012313450000000002fe T'FJMFTIF0'
```

TNS entries in processor RFTIF:

```
FJMFTIF0\
      TSEL  OSITYPE T'FJMFTIF0'
      TSEL  RFC1006 T'FJMFTIF0'
A01FTIF0\
      TSEL  OSITYPE T'A01FTIF0'
      TSEL  RFC1006 T'A01FTIF0'
ftifr2\
      TA    RFC1006 123.22.44.56 PORT 1100 T'$FJAM'
ftifr1\
  TA   OSITYPE 470058+0144450100000123134500000000013fe T'$FJAM'
```

TNS entries in processor R2:

```
$FJAM\
      TSEL    RFC1006 T'$FJAM'
$FJAMOUT\
      TSEL    RFC1006 T'$FJAMOUT'
ftr1\
      SSEL    A'FTIFR1'
      TA      RFC1006 123.22.33.45  T'FJMFTIF0'
```

8.3.4 Definition of remote TS applications for *open*FT-FTAM

For each FTAM system which is to be accessible for requests issued locally, or for which the automatic restart is to be provided, a TNS entry must be made. For FTAM partners, you must specify the presentation and session selectors. This only works for CMX V5.0 when you are using *tnsxcom* or the CMX GUI. The presentation/session and transport selector entries can be made in ASCII (A'...'), EBCDIC (E'...'), TRANSDATA format (T'...') or hexadecimal (X'...'). Presentation and session selectors may only be between 0 and 16 bytes long. If the presentation or session selector is missing, the entries `PSEL V''` or `SSEL V''` are absolutely necessary. With transport addresses for FTAM partners, no CC list may be specified.

Special points

- With the TCP/IP-LAN transport system in the local system, you must enter the Internet address, the transport selector, as well as the port number of the partner processor. RFC1006 partner systems which support port 102 (e.g. BS2000/OSD and UNIX systems with CMX V5.0) are assigned the port number 102; all other partner systems are given the port number specified in the particular partner system for the $FTAM application.
- The entries of the file to be translated with *tnsxcom* must in principle look the same as in the following examples on page 203. You can use the following checklist to assist you.

Checklist

The following checklist is intended to help you gather the data required for the TNSX entry of an FTAM partner. The questions must be answered by the FTAM partner.

1. openFT-FTAM sets up the connection.

   Which values do the following parameter have (with specification of coding):

a)	called X121/ LAN address/ NSAP/X.31	_____		
b)	called TSEL	_____	Code:	_____
c)	called SSEL	_____	Code:	_____
d)	called PSEL	_____	Code:	_____

e)	Protocol Identifier (Layer 3 CUD)	_____
f)	called APT	_no _____NILAPTitle __ [1)]
g)	called AEQ	_no _____ [1)]
h)	calling APT	_no _____NILAPTitle __ [1)]

[1)] APT (Application Process Title) and AEQ (Application Entity Qualifier) are not specified in the TNS entries, but in the *open*FT commands. Some FTAM partners expect APTs and possibly AEQs; others expect no APTs/AEQs to be specified.

2. The partner sets up the connection.

   Which values do the following parameters have (with specification of coding):

a)	calling X121/ LAN address/ NSAP/X.31	_____		
b)	calling TSEL	_____	Code:	_____
c)	calling SSEL	_____	Code:	_____
d)	calling PSEL	_____	Code:	_____

You must observe correct notation (uppercase and lowercase) and remember that blanks and X‘00‘ must be specified correctly for selectors.

8.3.4.1 Sample entries for FTAM partners

- Entries for a link to processor ***BLUE*** via X.25
  - The partner requires the selectors in ASCII format. It does not require a protocol identifier.

```
blue\
   PSEL    A'FTAMBLUE'
   SSEL    A'FTAMBLUE'
   TA      WANSBKA 45890000001 A'FTAMBLUE'
```

  - The following entry is necessary when processor *BLUE* has the initiative. It is used only to identify the initiator (sender check).

```
blue#1\
   PSEL    A' '
   SSEL    A'P'
   TA      WANSBKA 45890000001 A'@'
```

- Entries for a link to processor *DEX* via X.25
  - The partner requires the selectors in ASCII format, it does not require a protocol identifier. The partner just sends empty selectors when it has the initiative.

```
dex\
   PSEL    A'TS'
   SSEL    A'TS-SSAP'
   TA      WANSBKA 45890000001 A'TS-TSAPEAF' X'03010100'
```

  - The following entry is necessary when processor *DEX* has the initiative. It is used only to identify the initiator.

```
dex#1\
   PSEL    V''
   SSEL    V''
   TA      WANSBKA 45890000001 V''   X'03010100'
```

- Entry of a partner address for transfer via TCP/IP-RFC1006. The partner supports the standardized port number 102 of RFC1006.

```
ftamrfc\
     PSEL    V''
     SSEL    V''
     TA      RFC1006  123.4.5.67      T'$FTAM'
;                     Internet addr.  T selector
```

- Entry of a partner address (*open*FT for Windows with FTAM functionality) for transfer via TCP/IP-RFC1006 (Port 4800) :

```
ftamwnt\
    PSEL    V''
    SSEL    V''
    TA      RFC1006  123.4.5.68     PORT 4800    A'SNI-FTAM'
;                    Internet addr  Portno       T selector
```

- Entry of a partner address for transfer via ETHN-CLNS/active:

```
ftametha\
    PSEL    V''
    SSEL    V''
    TA      OSITYPE 49+006C080015304050FE T'$FTAM'
 ;                       OSI network addr. T selector
```

(OSI network address as per ISO Standard 8348/Add.2; the structure is described in the CMX manual.)

- Entry of a partner address for transfer via ETHN-CLNS/passive:

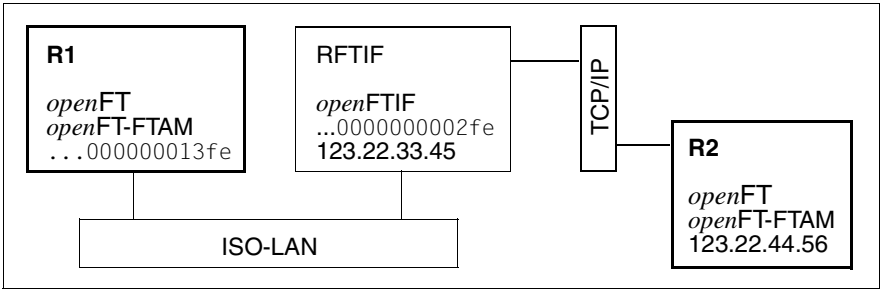
```
ftamethp\
    PSEL    V''
    SSEL    V''
    TA      LANSBKA 080014110960  T'$FTAM'
 ;                  Ethernet addr.T selector
```

- Entry of a partner address for transfer via WAN-CONS, ISDN-CONS

```
ftamcons\
    PSEL    V''
    SSEL    V''
    TA      WANSBKA X.121 45890040034 T'$FTAM' X'D5000002'
 ;                        SNPA info    T sel.   TPI
```

8.3.4.2 *open*FTIF sample for linking UNIX systems via FTAM protocol

In the following example, the two UNIX systems R1 and R2 are linked with the aid of an RFTIF gateway processor (with *open*FTIF for UNIX systems software) via an ISO-LAN and a TCP/IP network. File transfer is possible in both directions. CMX V5.0 is installed in both processors.



All TNS entries in the processors R1, R2 and RFTIF required for file transfer between R1 and R2 are described.

TNS entries in processor R1:

```
$FTAM\
 PSEL   V''
 SSEL   V''
 TSEL   OSITYPE T'$FTAM'
ftamr2\
 SSEL  A'ftifr2'
 TA OSITYPE 470058+0144450100000123134500000000002fe T'FJMFTIF0‘
```

TNS entries in processor RFTIF:

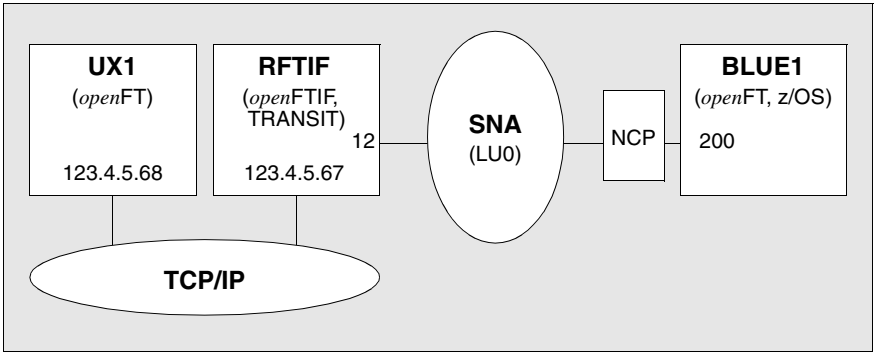
```
FJMFTIF0\
 TSEL   OSITYPE T'FJMFTIF0'
 TSEL   RFC1006 T'FJMFTIF0‘'
ftifr2\
 SSEL   V''
 TA     RFC1006 123.22.44.56  T'$FTAM'
ftifr1\
 SSEL   V''
 TA    OSITYPE 470058+0144450100000123134500000000013fe T'$FTAM'
```

TNS entries in processor R2:

```
$FTAM\
 PSEL    V''
 SSEL    V''
 TSEL    RFC1006 T'$FTAM'
ftamr1\
 PSEL    V''
 SSEL    A'ftifr1'
 TA      RFC1006 123.22.33.45  T'FJMFTIF0'
```

8.4 Example entries for linking with *open*FT for z/OS over *open*FTIF

In the following example, a UNIX system, which is connected to a TCP/IP network, is linked to an z/OS computer via the gateway computer RFTIF. The z/OS computer is connected to an SNA(LU0) network via a preprocessor (NCP).



UX1 and BLUE1 are the FTIF names of the two partner computers.

CMX, the TRANSIT-CLIENT and the TRANSIT-SERVER are installed on the gateway computer RFTIF. The main station of the computer is identified in the SNA network with LU number 12 and LU name FJMGW001. A maximum of 8 parallel connections are to be possible.

For this link, entries are required in the TNS of both UNIX systems, KOGS entries for TRANSIT as well as entries in the NCP generation and the network description file of the z/OS computer.

TNS entry in computer UX1

```
ftlu0\
      SSEL A'BLUE1'
      TA RFC1006 123.4.5.67 PORT 1400 T'FJMFTIF0'
```

If the entry PORT 1400 is omitted, port number 102 is used.

TNS entries in computer RFTIF

```
FJMFTIF0\
      TSEL LANINET A'1400'
      TSEL RFC1006 T'FJMFTIF0'
      TSEL EMSNA T'FJMGW001' 12

A01FTIF0\
      TSEL LANINET A'1401'
      TSEL RFC1006 T'A01FTIF0'
      TSEL EMSNA T'A01GW001' 13

...
...

A08FTIF0\
      TSEL LANINET A'1408'
      TSEL RFC1006 T'A08FTIF0'
      TSEL EMSNA T'A08GW001' 20

ux1\
      TA RFC1006 123.4.5.68 PORT 1100 T'$FJAM'

blue1\
      TA EMSNA T'FJMFTMVS' 0/0
```

FJMFTMVS is the name of the main station in the z/OS computer. FJMGW001 is the name with which the gateway application in the *open*FTIF computer is addressed from the z/OS. A01GW001 to A01GW008 are the subapplications which handle the requests.

All of these names must be generated in the VTAM of the z/OS.

KOGS entries in TRANSIT

The line to the z/OS computer (XLINK macro), the z/OS computer itself (XPU macro), the main station and all substations (LU macro) must be defined in the TRANSIT-SERVER. The appropriate menu can also be used instead of the macros.

```
XLINK   LO2CG8,
        ....
XPU     PO2CG8,
        TYP         = HOST,
        ...
        LINK        = LO2CG8
        ...
#
# LU for the openFTIF main station
XLU     FJMGW001,
```

```
        TYP=F,
        PUCONNECT   = APHSTART,
        CTYP        = PUBLIC,
        LOCADDR     = 12,
        PU          = PO2CG8,
        AUTO-LOGOFF = 0
#
# LUs for the 8 openFTIF substations
XLU A01GW001,
        TYP=F,
        PUCONNECT   = APHSTART,
        CTYP        = PUBLIC,
        LOCADDR     = 13,
        PU          = PO2CG8,
        AUTO-LOGOFF = 0
...
XLU A08GW001,
        TYP=F,
        PUCONNECT   = APHSTART,
        CTYP        = PUBLIC,
        LOCADDR     = 20,
        PU          = PO2CG8,
        AUTO-LOGOFF = 0
```

Entries in computer BLUE1

- You must make the following entry in the network description file:

  ```
  FJADDSYS FTUNIX,SYSADR=UX1,RELADR=FJMGW001
  ```

  FTUNIX is the partner name, which must be specified in an FT request. FJMGW001 is the LU name of the main station, which is linked to the LU number (here: 12) in the NCP generation (see below).

- You must generate the line to the RFTIF computer (LINE statement), the TRANSITSERVER (PU statement) and the gateway application with all subapplications (LU statement) in the NCP generation:

  ```
  *
  TRANSGRP GROUP
  *
  L48       LINE ADDRESS=(48,FULL),....
  *
  PU48      PU   ADDR=C1,           -
                 DISCNT=NO,         -
                 DLOGMOD=SNX32702,  -
                 IDBLK=017,         -
                 IDNUM=20008,       -
  ```

```
                ISTATUS=ACTIVE,    -
                MAXDATA=4105,      -
                MAXOUT=7,          -
                MAXPATH=2,         -
                MODTAB=MOD3270,    -
                PACING=4,          -
                PUTYPE=2,          -
                SSCPFM=USSSCS,     -
                USSTAB=USSSCS,     -
                VPACING=4
*
FJMGW001  LU    LOCADDR=12,        -
                PACING=3,          -
                VPACING=2,         -
                DLOGMOD=FJMLMOD,   -
                MODETAB=MODFTMSP,  -
                SSCPFM=FSS,        -
                USSTAB=ISTINCDT
A01GW001  LU   LOCADDR=13,         -
                PACING=3,          -
                VPACING=2,         -
                DLOGMOD=FJMLMOD,   -
                MODETAB=MODFTMSP,  -
                SSCPFM=FSS,        -
                USSTAB=ISTINCDT
...
A08GW001  LU    LOCADDR=20,        -
                PACING=3,          -
                VPACING=2,         -
                DLOGMOD=FJMLMOD,   -
                MODETAB=MODFTMSP,  -
                SSCPFM=FSS,        -
                USSTAB=ISTINCDT
```

- In VTAM or NCP, the main station FJMFTMVS and the 8 substations A01FTMVS to A08FTMVS must be generated:

```
FJMFTMVS APPL  ACBNAME=FJMFTMVS,  -
               AUTH=(ACQ,VPACE),  -
               DLOGMOD=FJMLMOD,   -
               MODETAB=modtab,    -
               PRTCT=ft-kennwort, -
               VPACING=3
A01FTMVS APPL  ACBNAME=A01FTMVS,  -
               AUTH=(ACQ,VPACE),  -
               DLOGMOD=FJMLMOD,   -
               MODETAB=modtab,    -
               PRTCT=ft-kennwort, -
               VPACING=3
```

```
...
A08FTMVS APPL  ACBNAME=A08FTMVS,  -
               AUTH=(ACQ,VPACE),  -
               DLOGMOD=FJMLMOD,   -
               MODETAB=modtab,    -
               PRTCT=ft-kennwort, -
               VPACING=3
```

8.5 *open*FT in a Cluster with UNIX based systems

Software requirements

The same version of *open*FT must be installed on all nodes of the cluster. In addition, the following communications software is required:

SUN Solaris (Sparc)	CMX version 5.1E50 and later
all platforms	PCMX version 4.1A10 and later

On SUN, TNS inputs are only allowed to contain TCP/IP components. An input file for the *tnsxcom* command could look like the following:

```
$FJAM      DEL

$FJAM\
   TSEL    RFC1006  T'$FJAM'    ; input for TCP/IP-RFC1006
   TSEL    LANINET  A'1100'     ; input for TCP/IP

$FJAMOUT   DEL

$FJAMOUT\
   TSEL    RFC1006  T'$FJAMOUT' ; input for TCP/IP-RFC1006
   TSEL    LANINET  A'1101'     ; input for TCP/IP

$FTAM      DEL

$FTAM
   PSEL    V''                  ; blank presentation selector
   SSEL    V''                  ; blank session selector
   TSEL    RFC1006  T'$FTAM'    ; input for TCP/IP-RFC1006
   TSEL    LANINET  A'4800'     ; input for TCP/IP
```

During this, the existing inputs in the TNS are overwritten by *tnsxcom*.

Example 1: a fail-safe instance

The cluster TREE (UNIX based systems, IP-address 123.25.10.12) consists of the two computers MAPLE (IP-address 123.25.10.1) and BEECH (IP-address 123.25.10.2). The failure management concept allows TREE to run either on MAPLE or BEECH. Only one openFT instance is fail-safe in this case.

Configure the cluster in such a way that a disk is always available. In this example, it is the directory */openFT*.

Required steps for the computer MAPLE

1. Install *open*FT (including the add-on products *open*FT CR *open*FT FTAM and *open*FT-FTP, if necessary)
2. Deactivate *open*FT:

   ```
   ftstop
   ```

3. Adapt the $FJAM and $FJAMOUT TNS inputs to Sun. They may only contain RFC1006 and LANINET inputs, see above.
4. Set the address for the instance *std*:

   ```
   ftmodi std -addr=MAPLE
   ```

5. Activate *open*FT on the instance *std* and set the ID, if this did occur automatically during installation:

   ```
   . ftseti std
   ftmodo [-id=MAPLE.FOREST.NET]
   ftstart
   ```

6. Mount the disk */openFT* on MAPLE.
7. Create the new instance *cluster* and check it. The directory */openFT* must exist, whereas the directory */openFT/cluster* must not exist:

   ```
   ftcrei cluster /openFT/cluster -addr=TREE.FOREST.NET
   ftshwi @a -l
   ```

8. If authentication is to be used in the instance *cluster*, then public keys from the partner systems must be stored in the directory */openFT/cluster/syskey*, or the public key from the directory */openFT/cluster/config* must be made available to the partner systems.
9. Deactivate the instance *cluster*:

   ```
   ftseti std; ftdeli cluster
   ```

Required steps on for the computer BEECH

1. Install *open*FT (including the add-on products *open*FT-CR *open*FT-FTAM and *open*FT-FTP, if necessary)
2. Deactivate *open*FT:

   ```
   ftstop
   ```

3. Adapt the $FJAM and $FJAMOUT TNS inputs if they exist. They may only contain RFC1006 and LANINET inputs, see above.
4. Set the address of the instance *std*:

   ```
   ftmodi std –addr=BEECH
   ```

5. Activate *open*FT on instance *std* and set the ID, if this did not occur automatically during installation:

   ```
   . ftseti std
   ftmodo [–id=BEECH.FOREST.NET]
   ftstart
   ```

6. Next, make a shell script for administering the instance that handles the events *start*, *stop,* and *check*. The script must be available and properly configured on the computers **MAPLE** and **BEECH.** It might look like the following:

   ```
   PAR=$1
   BIN=/opt/bin; export BIN
   INST=cluster
   OPENFTINSTANCE=$INST; export OPENFTINSTANCE
   case $PAR in
   start) OPENFTINSTANCE=std; export OPENFTINSTANCE
      $BIN/ftcrei $INST /openFT/$INST
      case $? in
         0|5) continue;;
         *) exit 1;;
      esac
      OPENFTINSTANCE=$INST; export OPENFTINSTANCE
      $BIN/ftstart 2>/dev/null
      case $? in
         0|180) exit 0;;
         *) exit 1;;
      esac;;
   ```

```
stop) $BIN/fttop 2>/dev/null
   case $? in
      0|181) continue;;
      *) exit 1;;
   esac
   OPENFTINSTANCE=std; export OPENFTINSTANCE
   $BIN/ftdeli cluster
   case $? in
      0) exit 0;;
      *) exit 1;;
   esac;;
check) VALUE=`$BIN/ftshwo -csv 2>/dev/null |fgrep FtStarted\
               |sed s/";"/" "/g`
   [ -z $VALUE ] && exit 1
   set $VALUE
   i=1
   FTROW=1
   while [ "$1" != "FtStarted" ]
   do shift
   FTROW=`expr $FTROW + 1`
   done
   FTSTART=`$BIN/ftshwo -csv|fgrep -v FtStarted|cut \
           -f$FTROW -d\;`
   if [ $FTSTART = '*NO'  ]
   then # openFT server not active
   exit 1
   else # openFT server active
   exit 0
   fi
   ;;
esac
```

Working with individual instances

When everything is finished, there is a standard instance on both the MAPLE and BEECH computers which is not fail-safe. By making a selection on the graphical user interface, or by executing the command *ftseti std,* you will be working with the respective standard instance. You can make use of all the *open*FT functions in the standard instances (e.g. set up admissions profiles, view log records, etc.). The standard instances on MAPLE and BEECH can be addressed normally from external systems using the addresses of these computers (123.25.10.1 or 123.25.10.2).

The fail-safe instance *cluster* is available on one of these two computers; the one on which the disk */openFT* is currently mounted. You can work with the instance on this computer using the graphical user interface or by using the command . *ftseti cluster* and use all of *open*FT functions available there. It is not necessary to know on which computer the disk */openFT* is mounted during this. You must choose TREE as the partner. The cluster TREE (*open*FT instance *cluster*) is addressed externally under the IP address 123.25.10.12.

Example 2: Fail-safe capability for both computers in the cluster

The cluster of UNIX systems, once again, consists of two computers: MAPLE (IP address 123.25.10.1) and BEECH (IP address 123.25.10.2).

In this example, however, there is to be a fail-safe *open*FT instance available on each of the two computers. For this purpose, the computers are superimposed (MAPLE by CL_MAPLE (IP address 123.25.10.10) and BEECH by CL_BEECH (IP address 123.25.10.20). If the computer MAPLE fails, then CL_MAPLE is switched over to the computer BEECH. If the computer BEECH fails, then CL_BEECH is switched over to the computer MAPLE.

Configure the cluster so that a disk is always available for each computer, for example: */sha_MAPLE* and */sha_BEECH*.

Required steps for the computer MAPLE

1. Configure a standard instance as shown in example 1.
2. Mount the disk */sha_MAPLE* and */sha_BEECH* on MAPLE.
3. Create and check the instances *MAPLE* and *BEECH*:

```
ftcrei MAPLE /sha_MAPLE/oFT -addr=CL_MAPLE.FOREST.NET
ftcrei BEECH /sha_BEECH/oFT -addr=CL_BEECH.FOREST.NET
ftshwi @a -l
```

4. Deactivate the instances *MAPLE* and *BEECH*:

```
ftdeli MAPLE
ftdeli BEECH
```

Required steps on the computer BEECH

1. Configure a standard instance as shown in example 1.
2. Next, make a shell script for controlling openFT on the computers MAPLE and BEECH that handles the events *start, stop, and check*. Both scripts must be available on both computers. The shell script might look like the example below (in the script for BEECH, the name *MAPLE* must be substituted with *BEECH* in the following):

```
PAR=$1
BIN=/opt/bin; export BIN
INST=MAPLE
OPENFTINSTANCE=$INST; export OPENFTINSTANCE
```

```
case $PAR in
start) OPENFTINSTANCE=std; export OPENFTINSTANCE
   $BIN/ftcrei $INST /sha_MAPLE/oFT
   case $? in
      0|5) continue;;
      *) exit 1;;
   esac
   OPENFTINSTANCE=$INST; export OPENFTINSTANCE
   $BIN/ftstart 2>/dev/null
   case $? in
      0|180) exit 0;;
      *) exit 1;;
   esac;;
stop) $BIN/ftstop 2>/dev/null
   case $? in
      0|181) exit 0;;
      *) exit 1;;
   esac
   OPENFTINSTANCE=std; export OPENFTINSTANCE
   $BIN/ftdeli $INST
   case $? in
      0)exit 0;;
      1)exit1;;
   esac;;
check) VALUE=`$BIN/ftshwo -csv|fgrep FtStarted \
               |sed s/";"/" "/g`
   set $VALUE
   i=1
   FTROW=1
   while [ "$1" != "FtStarted" ]
   do shift
      FTROW=`expr $FTROW + 1`
   done
   FTSTART=`$BIN/ftshwo -csv|fgrep -v FtStarted \
            |cut -f$FTROW -d\;`
   if [ $FTSTART = '*NO' ]
   then # openFT server not active
      exit 1
   else # openFT server active
      exit 0
   fi;;
esac
```

Working with the individual instances

When everything is finished, there is a standard instance on both the MAPLE and BEECH computers which is not fail-safe. By making a selection on the graphical user interface, or by executing the command *ftseti std,* you will be working with the respective standard instance. You can make use of all the *open*FT functions in the standard instances (e.g. set up admissions profiles, view log records, etc.). The standard instances on MAPLE and BEECH can be addressed normally from external systems using the addresses of these computers (123.25.10.1 or 123.25.10.2).

The *open*FT instances *MAPLE* and *BEECH* are available on the computer, on which the corresponding disk is currently mounted. They can be used, as usual, via the graphical user interface or the command interface. Another possibility exists using remote administration (where the partner is CL_MAPLE or CL_BEECH).

In order to transfer files to these instances, the IP addresses of CL_MAPLE.FOREST.NET or CL_BEECH.FOREST.NET (123.25.10.10 or 123.25.10.20) can be addressed.

8.6 Exit codes and messages for administration commands

With the following messages, the value for *fthelp* must be increased by 1000 1000, e.g. 1034 instead of 34. The description has the following format:

exit code Message text
meanings and measures as appropriate

20 openFT already started

Meaning:
*open*FT can only be started once in each instance.

Measure:
Terminate *open*FT if necessary.

21 Request must be canceled without FORCE option first

Meaning:
Before the FORCE option is used, the command must be called without the FORCE option.

Measure:
Issue the command without the FORCE option first.

29 Maximum number of key pairs exceeded

Meaning:
The maximum number of key pair sets has been reached.

Measure:
Before new key pair set can be created, an older key pair set must be deleted.

30 Warning: last key pair deleted

Meaning:
The last key pair set has been deleted. Without a key pair set, encrypted transfer, authentication and data integrity checking are not possible.

Measure:
Create a new key pair set.

31 No key pair available

Meaning:
All transfers are carried out without encryption.

Measure:
Create a new key pair set, if necessary.

32 Last key pair must not be deleted

33 The public key files could not be updated

Meaning:
The contents of the *syspkf* file could not be fully updated.
Possible reasons:
– The *syspkf* file is locked.
– There is not enough disk space to allow the file to be created.

Measure:
Take the appropriate action depending on the cause of the error:
– Unlock the file.
– Allocate disk space or have your system administrator do it.

Update the key with *ftupdk*.

34 Command only permissible for FT or FTAC administrator

Meaning:
Only the FT or FTAC administrator is permitted to use the command.

Measure:
Have the command executed by the FT or FTAC administrator.

35 Command only permissible for FT administrator

Meaning:
Only the FT administrator is permitted to use the command.

Measure:
Have the command executed by the FT administrator.

36 User not authorized for other user Ids

Meaning:
The user is not authorized to use a different user ID in the command.

Measure:
Specify your own ID, or have the command executed by the FT or FTAC administrator.

37 Key reference unknown

Meaning:
The specified key reference is unknown.

Measure:
Repeat the command with an existing key reference.

38 Request <request id> is in the termination phase and can no longer be canceled

39 openFT not active

Meaning:
*open*FT is not started.

Measure:
Start *open*FT, if necessary.

40 Config user ID unknown or not enough space

Meaning:
The Config user ID of the current instance is unknown or the disk space allocated is insufficient to allow creation of the request file, the file for storing trace data, or the key files.

Measure:
Either create the Config user IDor increase its disk space allocation or have your system administrator do it.

41 Specified file is not a valid trace file

42 openFT could not be started

43 Partner with same attribute <attribute> already exists in partner list

Meaning:
There is already a partner entry with the same attribute <attribute> in the partner list.

Measure:
The attribute <attribute> in partner entries must be unique. Correct the command accordingly and try again.

44 Maximum number of partners exceeded

Meaning:
The partner list already contains the maximum permissible number of partner entries.

Measure:
Delete partners that are no longer required.

45 No partner found in partner list

Meaning:
A partner for the specified selection could not be found in the partner list.

Measure:
Check if the specified partner name or address was correct.
If necessary, repeat the command using the correct name or address.

46 Modification of partner protocol type not possible

Meaning:
The protocol type of the partner entry cannot be changed subsequently.

Measure:
Delete the partner from the partner list, if necessary, and enter it again with a new protocol type.

47 Request <request id> not found

Meaning:
The request with the transfer ID <request id> could not be found.

Measure:
Specify the existing transfer ID and repeat the command.

8.7 Commands supported for the last time

This version is the last version in which the commands *fta*, *ftc* and *fti* will be supported.

ftc is replaced by *ftcanr* and *fti* is replaced by *ftshwo* and *ftshwr*. All these commands are described in the *open*FT User Guide.

fta - Administer *open*FT

i The *fta* command is supported for the last time in this version. Please use the following commands instead:
- *ftstart* and *ftstop* to start and stop the asynchronous *open*FT server
- *ftmodo* to modify the operating parameters
- *ftcrek* and *ftdelk* to generate and delete key record pairs.

The *-iq* parameter is not supported and the parameters *-sd* and *-d* have no effect.

Using *fta*, you can set operating parameters for *open*FT, start and exit the asynchronous *open*FT server, create new keys for encrypted data transfer and switch on and off trace mode for error diagnosis.

You can define the maximum number of asynchronous requests *open*FT is to execute simultaneously. You can also define the maximum length of the blocks to be transferred, and the range of file transfer requests to be logged by *open*FT.

If the asynchronous *open*FT server has not started, then *open*FT only processes synchronous requests and stores locally submitted asynchronous requests in the request queue, inbound requests are also not processed.

All *open*FT parameters are stored in a disk file. They are thus available in their original form the next time the system is started up.

Format

fta -h |
[-s | -t]
[-k] [-dk=<key reference 1..9999999>]
[-n | -f]
[-kl=0 | 768 | 1024] [-sd=n | y]
[-u=<transport unit size 512..65535>]
[-o=<maxosp 0..200>] [-i=<maxisp 0..200>]
[-p=<processor name 1..8>] [-l=<station name 1..8>]
[-id=<identification 1..64>]
[-ql=<Request lifetime 1..400>]
[-co=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 13 | 14 | 15 | 16]
[-lt=a | f | n] [-lc=a | m | r]
[-d=m | c]
[-ae=y | n]

Description

-h Displays the command syntax on the screen. Entries after the *-h* are ignored.

-s The asynchronous *open*FT server is started. An *openFTd* process (formerly *fta*) is then executed.

When starting the asynchronous *open*FT server, the protective bit setting is set for all files which are created by inbound requests. The protective bit setting is taken from the shell under which the *fta -s* command is entered. For more details, see section “Setting the protection bit for newly created files” on page 22.

-t The asynchronous *open*FT server is exited.

Unconditional termination of all activities. All active connections are first cleared down.
Requests present in the request queue are processed normally after the asynchronous *open*FT server has restarted. Requests which were aborted on termination of the asynchronous *open*FT server are executed after restart, provided that the partner supports this function.

When the *fta -t* command has been issued, the asynchronous *open*FT server can only be started again if all server processes are terminated. This may take some time, for example, if the connection cleardown is delayed by line problems.

-k Using this switch, you can create a new key pair set at any time.

-dk=<key reference 1..99999999>
Using this switch, you can delete the key pair set with the specified reference. After this, your system can no longer be authenticated by partner systems which are still using the accompanying public key. If you delete the last key pair set in your system, your *open*FT can no longer encrypt either request data or file content.

-n The monitoring function is activated. When activated, the diagnostic data are written to the trace files located in the directory */var/openFT/instance/traces* and (when linked to *open*FT partners) can be evaluated with the *fttrace* service program. Here *instance* means the name of the corresponding instance. A description of the evaluation of trace files is given starting on page 174.

-f The monitoring function is deactivated.

-kl=0 | 768 | 1024
The *-kl* parameter can be used to change the length of the RSA key used in encryption. The value of the *kl* parameter specifies the new RSA key length in bits. The RSA key is only used for the encryption of the AES key agreed between the partners (or for encrypting the DES key in versions up to *open*FT V7.0).
*open*FT uses the AES key for encrypting request description data and any file content present.
Encryption can be explicitly disabled using kl=0. The *fta -kl=...* command can be specified in current *open*FT operation.

When the request queue is created, *kl=768* is used as the default value.

-sd=n | y
This option is only supported for reasons of compatibility. Any specifications that are made are ignored since an SNA connection is implemented via *open*FTIF.

-u=transport unit size
Defines the maximum length of the blocks to be transferred within the range 512 up to maximum value of 65535. This upper limit is required, since the NEABF protocol elements SAC and SDK are not fragmentable and the maximum possible length of these protocol elements has increased as a result of the increased *open*FT key lengths (up to 1024 bits).

The default is 65535 characters.

The transport unit size is only valid for requests involving *open*FT partners.

-o=maxosp
-i=maxisp
Maximum number of locally submitted asynchronous outbound requests (*maxosp*) and remotely submitted inbound requests (*maxisp*) that can be processed simultaneously.

The default value for *-o* is 12
The default value for *-i* is 4

These two parameters can no longer be set explicitly using the new *open*FT V10 commands. They have been replaced by the connection limit (*ftmodo -cl*). *maxisp*, *maxosp* and the connection limit are related as follows:

– If the connection limit is defined using *ftmodo -cl* then *maxisp* and *maxosp* are derived from this value using a specific algorithm.

– If *maxisp* and *maxosp* are set explicitly using *fta* then the connection limit is the total of the two values.

The following restriction applies to FTAM partners: the maximum number of connections that can be active simultaneously corresponds to half the number of the files that a process can open simultaneously.

The total of *maxosp* and *maxisp* may not exceed 255.

-p=processor name
You specify the processor name assigned to your system here.

-l=station name
The station name of the *open*FT application. The default value is $FJAM.

The specifications for *processor name* and *station name* depend on how your system is connected to the network. Further details can be found in the chapter chapter “Installation and configuration” on page 43.

-id=identification
Specifying the instance identification of your *open*FT instance. Partner systems using *open*FT Version 8.1 and later, address your system via this string. In return, *open*FT uses the instance ID as the sender address when addressing the partners. The instance ID must be unique and not case-sensitive (see also section “Instance Identifications” on page 27). If you modify the instance ID, the relevant public key files will be automatically updated.

-ql=Request lifetime

Here you specify the maximum lifetime of entries in the request queue (in days); the default value is 30, but any value between 1 and 400 days is permitted. Both outbound and inbound requests in the request queue are deleted after the specified time span. In the case of outbound requests, this value can be combined with the *-ct*=... option in the *ft* command.

-co=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 13 | 14 | 15 | 16

This option is used to set a new character set, which is represented by a code table. The default code table is ISO 8859-1; this setting can be modified if required by specifying a numerical value for -co in the corresponding variant of the ISO 8859 character set.

The code table specification is only valid for *open*FT requests. If the partner system does not support the code table you used, the request will be canceled and a corresponding error message will be issued.

-lt=a | f | n

This option is used to selectively deactivate FT log records.
With connections to FTAM partners, it can take up to a minute for the selection to become active.

a (all)

This is the default setting; log records are written for all FT requests.

f (failure case)

Log records are written for failed FT requests only.

n (none)

No log records are written.

-lc= a | m | r

This option is used to selectively deactivate FTAC log records.
With connections to FTAM partners, it can take up to a minute for the selection to become active.

a (all)

This is the default setting; log records are written for all FTAC access checks.

m (modifying FM calls)

Log records are written for all FTAC access checks on modifying file management requests issued by the remote system, and on rejected FTAC access checks.

r (reject case)
Log records are written for rejected FTAC access checks only.

-d= m | c
This option is only supported for reasons of compatibility. Any specifications that are made are ignored.

-ae=y | n
This option activates/deactivates the AET (Application Entity Title).

y A "nil Application Entity Title" is included as the calling or called Application Entity Title (AET) for transfer using the FTAM protocol.

n The AET is deactivated. The option only has to be reset to *-ae=n* if FTAM link partners, as responders, do not expect to receive an AET.

Glossary

Italic type indicates a reference to other terms in this glossary.

absolute path name

The entire path name, from the root directory to the file itself.

access control

File attribute in the *virtual filestore*, attribute of the *security group* that defines *access rights*.

access protection

Comprises all the methods used to protect a data processing system against unauthorized system access.

access right

Derived from the *transfer admission*. It defines the scope of access for the user who specifies the transfer admission.

action list

Component of the file attribute *access control* (attribute of the *security group*) in the *virtual filestore* that defines *access rights*.

admission profile

Way of defining the *FTAC* protection functions. Admission profiles define a *transfer admission* that has to be specified in *FT requests* instead of the *LOGON or Login authorization*. The admission profile defines the *access rights* for a user ID by restricting the use of parameters in *FT requests*.

admission profile, privileged

-> see *privileged admission profile*

admission set

In *FTAC*, the admission set for a particular user ID defines which FT functions the user ID may use and for which *partner systems*.

admission set, privileged

-> see *privileged admission set*

AES (Advanced Encryption Standard)

The current symmetrical encryption standard, established by NIST (National Institute of Standards and Technology), based on the Rijndael algorithm, developed at the University of Leuven (B).

ANSI code

Standardized 8-bit character code for message exchange. The acronym stands for "American National Standards Institute".

API (Application Program Interface)

An interface that is freely available to application programmers. It provides a set of interface mechanisms designed to support specific functionalities.

Application Entity Title (AET)

The Application Entity Title consists of Layer 7 addressing information of the *OSI Reference Model*. It is only significant for *FTAM partners*.

asynchronous request

Once the *FT request* has been submitted, it is processed independently of the user. The user can continue working once the system has confirmed acceptance of the request. (see also *synchronous request*).

authentication

- Security service that validates a predefined identity.
- Process used by *open*FT to check the unique identity of the request partner.

background process

A process that runs independently of the user process. A background process is started by placing the special character & at the end of a command. The process which initiates the background process is then immediately free for further tasks and is no longer concerned with the background process, which runs simultaneously

basic functions

Most important file transfer functions. Several basic functions are defined in the *admission set* which can be used by a login name. The six basic functions are:
– inbound receive
– inbound send
– inbound follow-up processing
– inbound file management
– outbound receive
– outbound send

character repertoire

Character set of a file in the *virtual filestore.*

In the case of files transferred with *FTAM partners* it is possible to choose between: **GeneralString, GraphicString, IA5String and VisibleString**

cluster controller

Device for the connection between a transmission line and several terminals (data terminal).

Device for connecting a data communication line and a number of devices.

Comma Separated Value (CSV)

This is a quasi-tabular output format that is very widely used in the PC environment in which the individual fields are separated by a semicolon ";". It permits the further processing of the output from the most important *open*FT commands using separate tools.

communication controller

-> see *preprocessor*

compression

This means that several identical successive characters can be reduced to one character and the number of characters is added to this. This reduces transfer times.

computer network, open

-> see *open computer network*

concurrency control

Component of the FTAM file attribute *access control* (part of the *security group*) in the *virtual filestore* that controls concurrent access.

connectivity

In general, the ability of systems and partners to communicate with one another. Sometimes refers simply to the communication possibilities between transport systems.

constraint set

Component of the *document type*.

contents type

File attribute in the *virtual filestore,* attribute of the *kernel group* that describes the file structure and the form of the file contents.

data communication system

Sum of the hardware and software mechanisms which allow two or more communication partners to exchange data while adhering to specific rules.

data compression

Reducing the amount of data by means of compressed representation.

data encoding

Way in which an *FT system* represents characters internally.

Data Encryption Standard (DES)

International data encryption standard for improved security. The DES procedure is used in the FT products of Fujitsu Siemens Computers to encrypt the request description data and possibly the request data.

data protection

- In the narrow sense as laid down by law, the task of protecting personal data against misuse during processing in order to prevent the disclosure or misappropriation of personal information.
- In the wider sense, the task of protecting data throughout the various stages of processing in order to prevent the disclosure or misappropriation of information relating to oneself or third parties.

data security

Technical and organizational task responsible for guaranteeing the security of data stores and data processing sequences, intended in particular to ensure that
– only authorized personnel can access the data,
– no undesired or unauthorized processing of the data is performed,
– the data is not tampered with during processing,
– the data is reproducible.

DHCP

Service in TCP/IP networks that automatically assigns IP addresses and TCP/IP parameters to clients on request.

directory

In the tree-like UNIX file system or BS2000 (POSIX), directories exist in the form of lists that refer to files and to other directories.

In BS2000 (DVS), PLAM libraries are interpreted as directories.

document type

Value of the file attribute *contents type* (attribute of the *kernel group*). Describes the type of file contents in the *virtual filestore*.
– *document type* for text files: FTAM-1
– *document type* for binary files: FTAM-3

EBCDIC

Standardized code for message exchange as used in BS2000/OSD. The acronym stands for "Extended Binary Coded Decimal Interchange Code".

emulation

Components that mimic the properties of another device.

entity

-> see *instance*

Explorer

A program from Microsoft that is supplied with Windows operating systems to facilitate navigation within the file system.

file attributes

A file's properties, for example the size of the file, access rights to the file or the file's record structure.

file management

Possibility of managing files in the remote system. The following actions are possible:
– Create directories
– Display and modify directories
– Delete directories
– Display and modify file attributes
– Rename files
– Delete files.

filestore, virtual

-> see *virtual filestore*

file transfer

File transfer

file transfer request

-> see *FT- request*

firewall processor

Processor which connects two networks. The possible access can be controlled precisely and also logged.

fixed-length record

A record in a file all of whose records possess the same, agreed length. It is not necessary to indicate this length within the file.

FJAM-LU

FT-specific software module that is required for the connection of *open*FT for z/OS to a *TRANSDATA network* via *TRANSIT-SNA*. FJAM-LU is a component of TRANSIT-SNA.

follow-up processing

FT function that initiates execution of user-specified commands or statements in the *local* and/or the *remote system* after an *FT request* has been completed. The user may define different follow-up processing, depending on the success or failure of FT request processing. See also *preprocessing* and *postprocessing*.

follow-up processing request

Statements contained within an *FT request* for *follow-up processing* to be performed after file transfer.

FTAC (File Transfer Access Control)

Extended access control for file transfer and file management. In the case of BS2000 and z/OS, this is implemented by means of the product *open*FT-AC, for other operating systems it is a component of the *open*FT product, e.g. in *open*FT for UNIX systems.

FTAC administrator

Administrator of the FTAC functions; should be identical to the person responsible for data security in the UNIX system.

FTAC logging function

Function which FTAC uses to log each access to the protected system via file transfer.

FT administrator

Person who administers the *open*FT product installed on a computer. *open*FT can be administered from the login names *root* and *admin*.

FTAM-1

document type for text files

FTAM-3

document type for binary files

FTAM catalog

The FTAM catalog is used to extend the file attributes available in UNIX systems. It is only relevant for access using FTAM. For example, a file can be deleted using the command *rm* on a UNIX system, even if the *permitted actions* parameter does not allow this.

FTAM file attributes

All systems which permit file transfer via FTAM protocols must make their files available to their partners using a standardized description (ISO 8571). To this end, the attributes of a file are mapped from the physical filestore to a *virtual filestore* and vice versa. This process distinguishes between three groups of file attributes:

– kernel group: describes the most important file attributes.
– storage group: contains the file's storage attributes.
– security group: defines security attributes for file and system access control.

FTAM partner

Partner system that uses *FTAM protocols* for communication.

FTAM protocol (File Transfer, Access and Management)

Protocol for file transfer standardized by the "International Organization for Standardization" (ISO) (ISO 8571, FTAM).

FTIF (File Transfer Interconnect Facility)

Has the task of interconnecting different transport systems for file transfer; implemented in *open*FTIF for UNIX systems or Windows.

FTIF gateway

Computer on which *open*FTIF is installed.

FTIF name

Name used by *open*FTIF to identify the partner application in the destination system. This name is specified as a symbolic name (also referred to as GLOBAL NAME) for the partner application in a TNS entry in the FTIF gateway.

FTP partner

Partner system that uses *FTAM protocols* for communication.

FTP protocol

Manufacturer-independent protocol for file transfer in TCP/IP networks.

FT request

Request to an *FT system* to transfer a file from a *send system* to a *receive system* and (optionally) start *follow-up processing requests*.

FT system

System for transferring files that consists of a computer and the software required for file transfer.

FT trace

Diagnostic function that logs FT operation

functional standard

Recommendation defining the conditions and the forms of application for specific ISO standards (equivalent term: *profile*). The transfer of unstructured files is defined in the European Prestandard CEN/CENELEC ENV 41 204; file management is defined in the European Prestandard CEN/ CENELEC ENV 41205.

gateway

Generally understood to mean a computer that connects two or more networks and which does not function as a bridge. Variants: gateway at network level (=router or OSI relay), transport and application gateway.

gateway processor

Communication computer that links a computer network to another *computer network*. The mapping of the different protocols of the various *computer networks* takes place in gateway processors.

general string

Character repertoire for file files transferred to and from *FTAM partners*.

GraphicString

Character repertoire for files transferred to and from *FTAM partners*.

heterogeneous network

A network consisting of multiple subnetworks functioning on the basis of different technical principles.

homogenous network

A network constructed on the basis of a single technical principle.

HOSTS file

Network administration file that contains the Internet addresses, the processor names and the alias names of all accessible computers.

IA5String

Character repertoire for files transferred to and from *FTAM partners*.

identification

Procedure making it possible to identify a person or object.

inbound file management

Request issued in a *remote system* for which directories or file attributes of the *local system* can be displayed, file attribute modified or local file deleted.

inbound follow-up processing

Request issued in a *remote system* with *follow-up processing* in the *local system*.

inbound receive

Request issued in the *remote system*, for which a file is received in the *local system*.

inbound request / inbound submission

Request issued in another system, i.e. for this request.

inbound send

Request issued in a *remote system* for which a file is sent from the *local system*.

initiator

Here: *FT system* that submits an *FT request*.

instance / entity

A concept of OSI architecture: active element in a layer. Also see *openFT instance*.

instance ID

A network-wide, unique address of an *open*FT instance.

integrity

Unfalsified, correct data following the processing, transfer and storage phases.

interoperability

Capability of two *FT systems* to work together.

ISO/OSI reference model

The ISO/OSI Reference Model is a framework for the standardization of communications between open systems. (ISO=International Standards Organization).

job

Sequence of commands, statements and data

job transfer

Transfer of a file that constitutes a *job* in the *receive system* and is initiated as a job there.

kernel group

Group of file attributes of the *virtual filestore* that encompasses the kernel attributes of a file.

library

File with internal structure (elements)

library element

Part of a library. A library element may in turn be subdivided into a number of records.

Local Area Network (LAN)

Originally a high-speed network with limited physical extension. Nowadays, any network, that uses CSMA/CD, Token Ring or FDDI irrespective of the range **(see also** *WAN Wide Area Network*).

local system

The *FT system* at which the user is working.

logging function

Function used by *openFT* to log all file transfer accesses to the protected system.

logging record

Contains information about access checks performed by *openFT*.

Logical Unit (LU)

Interface between an application program and the SNA data communications network. The LU type describes the communications characteristics.

Login authorization

Transfer admission to a computer which (as a rule) consists of the login name and the password, and authorizes dialog operation, see also *LOGON authorization*.

LOGON authorization

Transfer admission authorizing access to a computer. The LOGON authorization (normally) consists of user ID, account number and password and authorizes the user to make use of interactive operation.

mailbox

The mailbox is a file which is read using the mail command. Each user has a mailbox for receiving messages.

maximum-string-length

Specifies the maximum length of *strings* within a file in the *virtual FTAM filestore*.

Network Control Program (NCP)

Operating system of the front-end-processor for SNA hosts.

network description file

File used up to *open*FT V9 that contains specifications concerning *remote systems* (*FT systems*).

open computer network

Computer network in which communication is governed by the rules of ISO/OSI. Interoperation of different computers from various vendors is made possible by defined *protocols*.

***open*FT-FTAM**

Add-on product for *open*FT (for BS2000, UNIX systems and Windows systems) that supports file transfer using FTAM protocols. FT-FTAM stands for File Transfer FTAM support

***open*FT-Script**

*open*FT interface providing a XML based script language that includes file transfer and file management functions. This interface allows you to combine several file transfer or file management requests to form a single *open*FT-Script request.

***open*FT-Script commands**

Commands used for administering *open*FT-Script requests.

***open*FTIF**

*open*FTIF performs the task of interconnecting different transport systems for file transfer.

***open*FT instance**

Several *open*FT systems, so-called *open*FT instances, can be running simultaneously on a cluster of a TCP/IP network . Each instance has its own address (instance ID) and is comprised of the loaded code of the *open*FT products (including add-on products if they are available) and of the variable files such as logging files, request log, etc.

***open*FT partner**

Partner system which is communicated with using *openFT protocols*.

***open*FT protocols**

Protocols standardized by Siemens AG for file transfer (SN77309, SN77312).

operating parameters

Parameters that control the *resources* (e.g. the permissible number of connections).

outbound request / outbound submission

Request issued in your own processor.

outbound receive

Request issued locally for which a file is received in the *local system*.

outbound send

Request issued locally for which a file is sent from the *local system*.

owner of an FT request

Login name in the *local system* or *remote system* under which this *FT request* is executed. The owner is always the ID under which the request is submitted, not the ID under which it is executed.

partner list

File containing specifications concerning *remote systems* (*FT systems*).

partner system

Here: *FT system* that carries out *FT requests* in cooperation with the *local system*.

password

Sequence of characters that a user must enter in order to access a user ID, file, job variable, network node or application. The user ID password serves for user *authentication*. It is used for access control. The file password is used to check access rights when users access a file (or job variable). It is used for file protection purposes.

PDN

Communication computer control program, consisting of the computer's operating system and system programs for the handling of communications protocols. Software that runs on a TRANSDATA data communications computer.

permitted actions

File attribute in the *virtual filestore*; attribute of the *kernel group* that defines actions that are permitted in principle.

Physical Unit (PU)

port number

Number that uniquely identifies a TCP/IP application or the end point of a TCP/IP connection within a processor.

POSIX (Portable Open System Interface)

Board and standards laid down by it for interfaces that can be ported to different system platforms.

postprocessing

*open*FT makes it possible to process the received data in the receiving system through a series of operating system commands, under the process control of *open*FT (in contrast to *follow-up processing*).

preprocessing

The preprocessing facility in *open*FT can be used to send a receive request in which the outputs of a remote command or program are transferred instead of a file. This makes it possible to query a database on a remote system, for example. Preprocessing also may be issued locally.

presentation

Entity that implements the presentation layer (layer 6) of the *ISO/OSI Reference Model* in an *FT system* that uses e.g. *FTAM protocols*.

presentation selector

Subaddress used to address a *presentation application*.

private key

Secret decryption key used by the recipient to decrypt a message that was encrypted using a *public key*. Used by a variety of encryption procedures including the *RSA procedure*.

privileged admission profile

Admission profile that allows the user to exceed the *FTAC administrator's* presettings in the *admission set*. This must be approved by the FTAC administrator who is the only person able to privilege admission profiles.

privileged admission set

Admission set belonging to the *FTAC administrator*. Exactly one admission set in the system has a privilege.

processor node

Entity in the host or communication computer that can be addressed throughout the network and that performs service functions for the exchange of data.

profile

In OSI, a profile is a standard which defines which protocols may be used for any given purpose and specifies the required values of parameters and options.

Here: a set of commands assigned to a user ID. The permissibility of these commands is ensured by means of syntax files. See also *admission profile*, *privileged admission profile*.

prompting in procedures

Function used to prompt the user at the terminal to enter data required to run the procedure.

protocol

Set of rules governing information exchange between peer partners in order to achieve a defined objective. This usually consists of a definition of the messages that are to be exchanged and the correct sequencing of messages including the handling of errors and other exceptions.

public key

Public encryption key defined by the receiver of a message, and made public or made known to the sender of the message. This allows the sender to encrypt messages to be sent to the receiver. Public keys are used by various encryption methods, including the *Rivest Shamir Adleman (RSA) procedure*. The public key must match the *secret key* known only to the receiver.

RAS

Remote Access Service; a Windows NT service that enables communication with remote systems.

receive file

File in the *receive system* in which the data from the *send file* is stored.

receive system

System to which a file is sent. This may be the *local system* or the *remote system*.

record

Set of data that is treated as a single logical unit.

relative path name

The path from the current *directory* to the file.

remote system

-> see *partner system*

request

Here: *FT request*

request file

File containing *asynchronous requests* and their processing statuses.

request identification / request ID

number that identifies an *FT request.*

request management

FT function responsible for managing *FT requests*; it ensures request processing from the submission of a request until its complete processing or termination.

request number

-> see *request identification*

request queue

File which contains the *asynchronous requests* and their processing states. The request queue also contains the parameters set with the *fta* command.

request storage

FT function responsible for storing *FT requests* until they have been fully processed or terminated.

resources

Hardware and software components needed by the *FT system* to execute an *FT request* (processes, connections, lines). These resources are controlled by the *operating parameters*.

responder

Here: *FT system* addressed by the *initiator.*

restart

Automatic continuation of an *FT request* following an interruption.

restart point

Point up to which the data of the *send file* has been stored in the *receive file* when a file transfer is interrupted and at which the transfer of data is resumed following a *restart*.

result list

List with information on a completed file transfer. This is supplied to the user in the *local system* and contains information on his or her *FT requests*.

RFC (Request for Comments)

Procedure used on the Internet for commenting on proposed standards, definitions or reports. Also used to designate a document approved in this way.

RFC1006

Supplementary protocol for the implementation of ISO transport services (transport class 0) using TCP/IP.

Rivest-Shamir-Adleman-procedure (RSA procedure)

Encryption procedure named after its inventors that operates with a key pair consisting of a *public key* and a *private key*. Used by FT products in order to reliably check the identity of the partner system and to transmit the AES key to the partner system for encrypting the request data.

router

Network element that is located between networks and guides message flows through the networks while simultaneously performing route selection, addressing and other functions. Operates on layer 3 of the OSI model.

security attributes

An object's security attributes specify how and in what ways the object may be accessed.

security group

Group of file attributes in the *virtual filestore*, encompassing the security attributes of a file.

security level

When *FTAC functions* areused, the security level indicates the required level of protection against a *partner system*.

send file

File in the *send system* from which data is transferred to the *receive file*.

send system

Here: *FT system* that sends a file. This may be the *local system* or the *remote system*.

server

Logical entity or application component which executes a client's requests and assures the (coordinated) usage of all the generally available services (File, Print, DB, Communication, etc.). May itself be the client of another server.

service

– As used in the OSI architecture: a service is the set of functions that a service provider makes available at a service access point.
– As used in the client/server architecture: a set of functions that a server makes available to its clients

service class

Parameter used by *FTAM partners* to negotiate the functions to be used.

session

– In OSI, the term used for a layer 5 connection.
– In SNA, a general term for a connection between communication partners
(applications, devices or users).

session selector

Subaddress used to address a *session* application.

shell metacharacters

The following metacharacters have special meanings for the shell:

*, [ ], ?, <, >, |, &, &&, ( ), { }

SNA network

Data communication system that implements the Systems Network Architecture (SNA) of IBM.

SNMP (Simple Network Management Protocol)

Protocol for TCP/IP networks defined by the Internet Engineering Task Force (IETF) for the transfer of management information.

special characters

-> see *shell metacharacters.*

standard error output (stderr)

By default, standard error output is to the screen.

standard input (stdin)

By default, standard input is from the keyboard.

standard output (stdout)

By default, standard output is to the screen.

storage group

File attribute in the *virtual filestore*, encompasses the storage attributes of a file.

string

Character string

string-significance

Describes the format of *strings* in files to be transferred using *FTAM protocols*.

synchronous request

The user process that submitted the *FT request* waits for transfer to terminate. The user cannot continue working (see also *asynchronous request*).

system

-> see *FT- system*

system, local

-> see *local system*

system, remote

-> see *remote system*

TCP/IP (Transmission Control Protocol / Internet Protocol)

Widely used data transmission protocol (corresponds approximately to layers 3 and 4 of the *ISO/OSI reference model*, i.e. network and transport layers); originally developed for the ARPANET (computer network of the US Ministry of Defense) it has now become a de-facto standard.

TRANSDATA network

Data communication system that implements the TRANSDATA network concept. Products used to connect *TRANSDATA networks* to *SNA networks* include, for example,
TRANSIT-CD and TRANSIT-SNA.

transfer admission

Authorization for file transfer and file management when using FTAC. The transfer admissions is then used in place of the *LOGON or Login authorization*.

transfer identification

-> see *request identification.*

transfer unit

In an FTAM environment, the smallest data unit for transporting file contents. For *FTAM-1* and *FTAM-3* these are *strings*. A transfer unit can, but need not, correspond to one file record.

TRANSIT-SERVER / TRANSIT-CLIENT

Products from Fujitsu Siemens Computers, used to link UNIX systems with *SNA networks*; running on the operating systems Reliant UNIX or Solaris.

TRANSIT-SNA

Product from Fujitsu Siemens Computers used to link *TRANSDATA networks* and *SNA networks*; running on the PDN operating system.

Transmission Control Protocol / Internet Protocol

-> see *TCP/IP*

transport connection

Logical connection between two users of the transport system (terminals or applications).

transport layer

Layer 4 of the *ISO/OSI reference model* on which the data transport protocols are handled.

Transport Name Service (TNS)

Service used to administer properties specific to transport systems. Entries for *partner systems* receive the information on the particular *transport system* employed.

transport protocol

Protocol used on the *transport layer*

transport selector (T-selector)

Subaddress used to address an ISO-8072 application in the *transport layer*

transport system

- The part of a system or architecture that performs approximately the functions of the four lower OSI layers, i.e. the transport of messages between the two partners in a communication connection.
- Sum of the hardware and software mechanisms that allow data to be transported in computer networks.

Unicode

The universal character encoding, maintained by the Unicode Consortium. This encoding standard provides the basis for processing, storage and interchange of text data in any language in all modern software and information technology protocols. The Unicode Standard defines three Unicode encoding forms:
UTF-8, UTF-16 and UTF-32.

universal-class-number

Character repertoire of a file in the *virtual filestore*.

variable length record

A record in a file all of whose records may be of different lengths. The record length must either be specified in a record length field at the start of the record or must be implicitly distinguishable from the next record through the use of a separator (e.g. Carriage Return - Line Feed).

virtual filestore

The FTAM virtual filestore is used by *FT systems* acting as *responders* to make their files available to their *partner systems*. The way a file is represented in the virtual filestore is defined in the FTAM standard, see *file attributes*.

visibleString

Character repertoire for files transferred to and from *FTAM partners*.

WAN (Wide Area Network)

A public or private network that can span large distances but which runs relatively slowly and with higher error rates when compared to a *LAN*. Nowadays, however, these definitions have only limited validity. Example: in ATM networks.

X terminal

A terminal or software component to display the graphical X Window interface of UNIX systems. An X terminal or a corresponding software emulation is a prerequisite for using the graphical interface of *open*FT.

Abbreviations

ACSE
Association Control Service Element

AES
Advanced Encryption Standard

AET
Application Entity Title

ANSI
American National Standards Institute

ASCII
American Standard Code for Information Interchange

BCAM
Basic Communication Access Method

BSFT
Byte Stream File Transfer

CAE
Common Application Environment

CEN
Comite Europeen de Normalisation

CENELEC
Comite Europeen de Normalisation Electrotechnique

CMX
Communication Manager SINIX

CCP
Communication Control Programm

DCAM
Data Communication Access Method

DCM
Data Communication Method

DES
Data Encryption Standard

DIN
Deutsches Institut für Normung (German standards institute)

DNS
Domain Name Service

EBCDIC
Extended Binary-Coded Decimal Interchange Code

ENV
Europäischer Normen-Vorschlag (European prestandard)

FADU
File Access Data Unit

FJAM
File Job Access Method

FSB
Forwarding Support Information Base

FSS
Forwarding Support Service

FT
File Transfer

FTAC
File Transfer Access Control

FTAM
File Transfer, Access and Management (ISO 8571)

FTIF
File Transfer Interconnect Facility

GPL
Gnu Public Licencse

GSM
Global System for Mobile Communication

ISAM
Index Sequential Access Method

ISO
International Organization for Standardization

LAN
Local Area Network

LMS
Library Maintenance System

MSV
Mittelschnelles Synchron Verfahren (Medium-fast synchronous method)

NDMS
Network Data Management System

NIS
Network Information Service

OSI
Open Systems Interconnection

OSS
OSI Session Service

PAM
Primary Access Method

PDN
Program system for data transmission and access control

PICS
Protocol Implementation Conformance Statement

PLAM
Primary Library Access Method

RFC1006
Request for Comments 1006

SAM
Sequential Access Method

SDF
System Dialog Facility

SNA
Systems Network Architecture

SNPA
Subnetwork Point of Attachment

TCP/IP
Transmission Control Protocol/Internet Protocol

TID
Transport Identification

TNSX
Transport Name Service in UNIX systems

TPI
Transport Protokoll Identifier

TS
Transport System

WAN
Wide Area Network

Related publications

The manuals are available as online manuals, see *http://manuals.fujitsu-siemens.com*, or in printed form which must be payed and ordered separately at *http://FSC-manualshop.com*.

***open*FT for UNIX systems**
Enterprise File Transfer in the Open World
User Guide

***open*FT for Windows**
Enterprise File Transfer in the Open World
User Guide
(only online available)

***open*FT for UNIX systems and Windows systems**
Program Interface
Programming Manua
(only online available)

***open*FT for UNIX systems and Windows systems**
***open*FT-Script Interface**
Programming Manua
(only online available)

***open*FT for BS2000/OSD**
Enterprise File Transfer in the Open World
User Guide

***open*FT for BS2000/OSD**
Installation and Administration
System Administrator Guide

***open*FT for BS2000/OSD**
Program Interface
Programming Manual

***open*FTIF for UNIX**
File Transfer Interconnect Facility with UNIX
User Guide

***open*FT for z/OS**
Enterprise File Transfer in the Open World
User Guide

***open*FT for z/OS**
Installation and Administration
System Administrator Guide

CMX
Operation and Administration
User Guide

CMX
Programming Applications
Programming Manual

OSS(SINIX)
OSI Session Service
User's Guide

X/Open CAE Specification
Byte Stream File Transfer (BSFT)
X/Open Document Number XO/CAE/91/400
X/OPEN Company Limited
November 1991

Index

D

S

Fujitsu Siemens Computers GmbH
User Documentation
81730 Munich
Germany

Fax: (++49) 700 / 372 00000

email: manuals@fujitsu-siemens.com
http://manuals.fujitsu-siemens.com

Comments
Suggestions
Corrections

Submitted by

Comments on *open*FT V10.0 for UNIX Systems
Installation and Administration

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *...@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *...@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.